

NASA Contractor Report CR-191135

**BRAYTON POWER CONVERSION SYSTEM PARAMETRIC DESIGN MODELLING FOR  
NUCLEAR ELECTRIC PROPULSION (TASK ORDER NO. 18)**

**Garrett Fluid Systems Division  
Allied-Signal Aerospace Company  
1300 W. Warner Road - P. O. Box 22200  
Tempe, Arizona 85282**

**Rocketdyne Division  
Rockwell International  
6633 Canoga Avenue  
Canoga Park, CA 91303**

November 21, 1993

**PREPARED FOR  
LEWIS RESEARCH CENTER  
UNDER CONTRACT NAS3 25808**

**NASA  
NATIONAL AERONAUTICS AND  
SPACE ADMINISTRATION**

## Forward

Systems engineering efforts initiated by NASA's Lewis Research Center (LeRC) in FY92 under RTOP 593-72, for Nuclear Electric Propulsion (NEP), have enabled the development of detailed mathematical (computer) models to predict NEP subsystem performance and mass. The computer models are intended to help provide greater depth to NEP subsystem (and system) modeling, required for more accurately verifying performance projections and assessing the impact of specific technology developments.

The following subsystem models have been developed:

- 1) liquid-metal cooled pin-type, and
- 2) gas-cooled NERVA (Nuclear Engine for Rocket Vehicle Applications) - derived for reactor/shield;
- 3) Potassium-Rankine, and
- 4) Brayton for power conversion;
- 5) heat rejection general model (includes direct Brayton, pumped loop Brayton, and shear flow condenser (Potassium Rankine);
- 6) power management and distribution (PMAD) general model, and
- 7) ion electric engine, and
- 8) magnetoplasmadynamic thruster for the electric propulsion subsystem.

These subsystem models for NEP were authored by the Oak Ridge National Laboratory (ORNL) for the reactor (NASA CR-191133), by the Rocketdyne Division of Rockwell International for Potassium Rankine (NASA CR-191134), and Brayton (NASA CR-191135) power conversion, heat rejection (NASA CR-191132), and power management and distribution (NASA CR-191136), and by Sverdrup Technology for the thrusters (NASA CR-191137).

At the time of this writing, these eight VAX/FORTRAN source and executable codes are resident on one of LeRC's Scientific VAX computers.

## CONTENTS

1.0 Summary .....	1
2.0 Brayton Power Conversion Modelling .....	4
2.1 Introduction .....	4
2.2 Parametric Design Code Description .....	4
2.2.1 Data Input/Setup Module .....	4
2.2.1.1 Thermodynamic Definition/CYC\$INP Input Data File	
2.2.1.2 Recuperator Input/REC\$INP Input Data File	
2.2.1.3 IHX Definition/IHX\$INP Input Data File	
2.2.1.4 Ducting Definition/DUCT\$INP Input Data File	
2.2.1.5 Subroutine BRAYTON	
2.2.1.6 Subroutine SETUP	
2.2.1.7 Subroutine GASPROP	
2.2.2 Thermodynamic Model .....	13
2.2.2.1 Subroutine STATEPT	
2.2.2.2 Subroutine BEARING	
2.2.2.3 Subroutine GENSIZE	
2.2.2.4 Subroutine WINDLOS	
2.2.2.5 Compressor Parametric Design Routines	
Subroutine AXCOMP	
Subroutine RADCOMP	
2.2.2.6 Turbine Parametric Design Routines	
Subroutines AXTURB and STRNGTH	
Subroutine RADTURB	
2.2.2.7 Gas Property Support Functions	
2.2.3 Postprocessor/Output Module .....	22
2.2.3.1 Subroutine BRAYTON	
2.2.3.2 Subroutine DUCTING	
2.2.3.3 Subroutine IHXSHEL	
2.2.3.4 Provisions for Calling the Heat Rejection Module	
2.2.3.5 Subroutine TAC	
2.2.3.6 Subroutine RECSIZE	
2.2.3.7 Subroutine OUTPUT	
2.2.4 Development of Generator Algorithms .....	25
2.2.4.1 Study Guidelines	
2.2.4.2 Generator Design Results	
2.2.4.3 Algorithm Development	
3.0 Conclusions and Recommendations .....	33

## **CONTENTS [continued]**

### **Appendices**

Appendix A - Summary of Global Nomenclature in Brayton System Model .....	34
Appendix B - Full Source Listing of Brayton Code .....	38
BRAY1.FOR .....	39
BRAY2.FOR .....	90
BRAY3.FOR .....	121
Appendix C - Input Data File Listings .....	141
CYC\$INP .....	142
REC\$INP .....	143
IHX\$INP .....	144
DUCT\$INP .....	145
Appendix D - Example Output [PROG\$OUT], 500 kWe Axial Case .....	146
Appendix E - Example Output [PROG\$OUT], 500 kWe Radial Case .....	149

## LIST OF FIGURES

1. Brayton Parametric Code - Top Level Flow Diagram .....	3
2. Brayton System Statepoint Locations .....	8
3. Axial Compressor Total-to-Static Polytropic Efficiency versus Specific Speed .....	16
4. Axial Compressor Tip Clearance Efficiency Correction .....	16
5. Axial and Radial Compressor Reynolds Number Efficiency Correction .....	17
6. Radial Compressor Total-to-Static Polytropic Efficiency versus Specific Speed .....	17
7. Radial Compressor Tip Clearance Efficiency Correction .....	18
8. Axial Turbine Total-to-Static Polytropic Efficiency versus Specific Speed .....	19
9. Axial Turbine Tip Clearance Efficiency Correction .....	20
10. Axial and Radial Turbine Reynolds Number Efficiency Correction .....	20
11. Radial Turbine Total-to-Static Polytropic Efficiency versus Specific Speed .....	21
12. Radial Turbine Tip Clearance Efficiency Correction .....	21
13. Recuperator Friction Factor and Heat Transfer Correlations .....	24

## LIST OF TABLES

1. CYC\$INP - Input Data File .....	5
2. Input Data File: CYC\$INP Nomenclature .....	6
3. Brayton Statepoint Locations .....	7
4. Example REC\$INP Data File .....	9
5. Example IHX\$INP Data File .....	10
6. Example DUCT\$INP Data File .....	11
7. Generator Design Requirements .....	26
8. Design Summary for Conventionally Wound TPTL Generators Operating at 700 ft/sec Surface Speed .....	29
9. Design Summary for Ring-Wound TPTL Generators Operating at 700 ft/sec Surface Speed .....	29
10. Design Summary for Ring-Wound TPTL Generators Operating at 500 ft/sec Surface Speed .....	30
11. Generator Materials and Technology Assessment .....	30



## 1.0 Summary

NASA LeRC is currently developing a Fortran based model of a complete NEP [nuclear electric propulsion] vehicle that would be used for piloted and cargo missions to the Moon and Mars. The proposed vehicle design will use either Brayton or K-Rankine [potassium-Rankine] power conversion cycle to provide the required electrical power. In support of the NEP modelling effort, Rocketdyne and Allied-Signal have teamed to generated parametric models of Brayton power conversion systems.

The Brayton system model is a thermodynamically based parametric design code which extensively uses "first principle" concepts, classical calculation methods, and simple (empirically based) algorithms to provide reasonable estimates of component size and performance levels. Realistic operating limits are also imposed on geometry, temperature, and speeds of system components.

The generator design algorithm defines a family of RW TPTL PMGs [ring-wound two-pole toothless permanent magnet generators] suitable for NEP applications. The algorithm was developed from a series of point designs and trade studies spanning the anticipated NEP range of requirements.

This documentation package was prepared by Rocketdyne and Allied Signal and is intended to facilitate use of the deliverable software written in support of NASA Task Order No. 18; Contract No NAS3-25808.

The software package supplied is comprised of the source codes, executable code, input data files, and output data file. The overall code consists of three source codes; BRAY1.FOR, BRAY2.FOR, and BRAY3.FOR. The source codes, compiled and linked together, form the executable BRAYTON.EXE code included in the package. The code requires four input data files which have been included in the software package. The data files are CYC\$INP, REC\$INP, IHX\$INP, and DUCT\$INP. The data contained in the input data files is typical geometry and performance data for a 500 kWe NEP Brayton system. The output data file, PROG\$OUT, contains the code output based on the input contained in the above input data files.

The code is annotated throughout, providing key information on methodology, units, and variable nomenclature where appropriate. Appendix A provides a listing of the global nomenclature used in the Brayton System model. Local nomenclature, e.g.; confined to a single routine, is found in the code listings included as Appendix B.

A complete listing of BRAY1.FOR, BRAY2.FOR, and BRAY3.FOR is included in Appendix B of this document. The listing of the input data files (CYC\$INP, REC\$INP, IHX\$INP, and DUCT\$INP) is provided in Appendix C. Two output (PROG\$OUT) examples from the code have been provided in Appendix D and E. Appendix D contains the output from a non-recuperated axial turbine/compressor case at 500 kWe. Appendix E contains the output from a recuperated radial turbine/compressor case at 500 kWe. These two cases have been provided as examples only and do not represent systems that have been optimized in any way. The computed parameters selected for output in PROG\$OUT represent a fraction of the computed parameters available from the code. The overall integrator of the codes has the option of using any of the parameters computed by the code when developing the overall code output.

It should be noted that the code, without exception, completes all computations assuming "engineering units". The input data files are all in SI units. The code reads the input data files and immediately converts the input to "engineering units" for computations. In the example output data file, the output has been provided in SI units. The parameters to be printed were converted from "engineering units" to SI units just prior to printing (or writing to file). Notations have been provided throughout the code to indicate the units that are in use.

Section 2 of this document discusses the software development of the Brayton system parametric design code. Figure 1 provides the top-level flow diagram for the code.

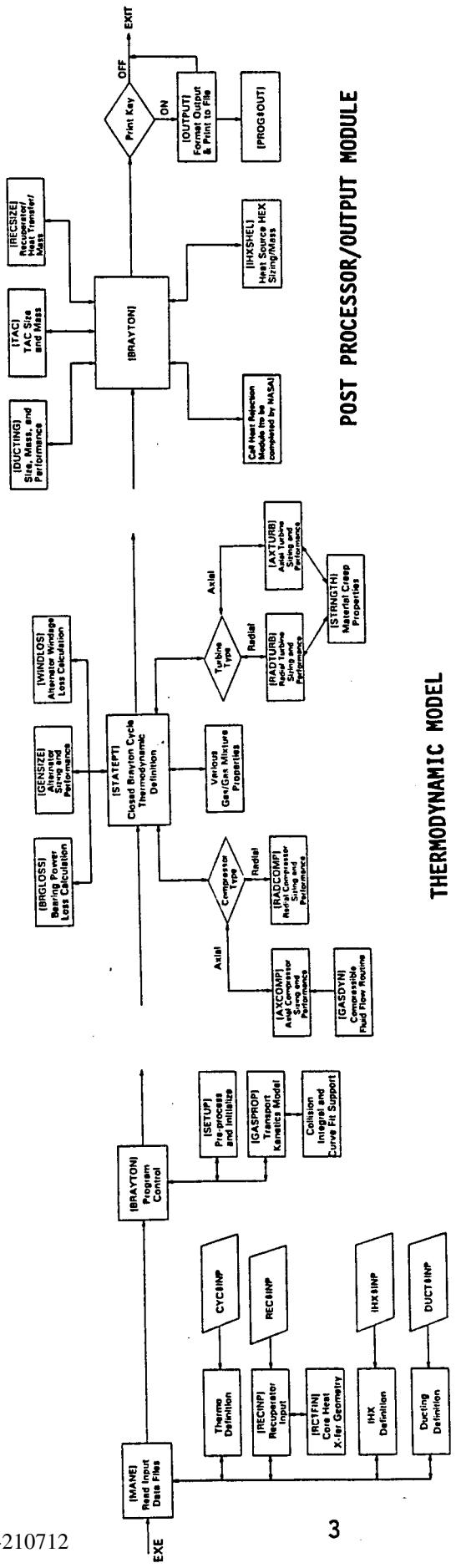


Figure 1. Brayton Parametric Code - Top Level Flow Diagram

## **2.0 Brayton Power Conversion Modelling**

### **2.1 Introduction**

The parametric design code is a system-level analysis which generates a thermodynamic cycle and companion parametric system/component designs based on specified input design variables. The code is particularly well suited to performing scoping evaluations and preliminary system-level parametric optimization.

The code, as currently configured, allows design of Brayton machines using Noble gases or any binary mixture of Noble gases, as the working fluid. However, the ducting and intermediate heat exchanger are currently configured to use only helium and xenon or any mixture of He/Xe. The ducting and intermediate heat exchanger codes would need to be upgraded in order to run the system code with a working fluid other than helium and xenon.

The thermodynamic and heat transfer internal structure of the code is fairly rigorous. Component design algorithms are somewhat less sophisticated due to the broad range of requirements and options available. The code is, by virtue of its thermodynamic central structure, highly amenable to refinement and added sophistication as specific NEP mission and requirements are defined.

### **2.2 Parametric Design Code Description**

The structure of the parametric design code, as shown in Figure 1, has been divided into three functional groups of modules; 1) Data Input/Setup Module, 2) Thermodynamic Module, and 3) Post Processor/Output Module. A description for each functional group of modules is provided in sections 2.2.1 - 2.2.3, respectively. Section 2.2.4 gives the details of the generator algorithm development.

#### **2.2.1 Data Input/Setup Module**

The deliverable software package is supplied with a temporary "mainline" program for use during familiarization and checkout. This code [Program MANE] can be replaced when integrating the routines into the overall system level code.

- Opens PROG\$OUT if IPRINT.GE.0
- Provides for the input of data from data files
- Calls BRAYTON subroutine

The code requires four small input data files; CYC\$INP, REC\$INP, IHX\$INP, and DUCT\$INP. The input data file CYC\$INP contains the required cycle and turbogenerator-compressor information. The input data file REC\$INP contains the recuperator core description. When the code is run without recuperation, the REC\$INP file is not required. The input data files IHX\$INP and DUCT\$INP contain the required Intermediate Heat Exchanger (IHX) and ducting descriptions, respectively.

### 2.2.1.1 Thermodynamic Definition/CYC\$INP Input Data File

As currently structured, a total of about twenty major independent variables are defined to establish a cycle description. These values are defined in input data file, CYC\$INP. As an example, a listing of the CYC\$INP data file is provided in Table 1.

Table 1. CYC\$INP - Input data file for cycle data

500 kWe CBC Test Case for NASA T018 Code - 5 December 92						
1	500.	.9	1400.	2.5	1144.44	
RADIAL	RADIAL	RING WOUND	TPTL	PMG	TRANSFORMER	DOWTHERM A
XENON	HELIUM	.5	511.11	522.22	.0	.0
5.	.005	.01	.01			
T1, K	PRC	MW	ER	NSC	N, RPM	DP/P6 DP/PREC
DP/P8	DP/P9	DP/P10	DP/P13	DP/P15	DP/P16	DP/P17
375.	1.8	20.	.85	42.	28000.	.005 .02
.005	.002	.003	.003	.005	.005	.005

The data contained in input file, CYC\$INP, is read in as follows:

```
READ (60,10) (CYCDES(I),I=1,8)
READ (60,20) IPRINT,POWER,PWRFCTR,VOLTAGE,GENASP,XTEMP(9)
READ (60,30) COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
READ (60,40) GAS(1),GAS(2),CPCLNT,XTINCLNT,XTOUTCLNT,XBLC,XBLT
READ (60,50) LIFETIME,FCTQ8,FCTQ10,FCTQ13
READ (60,10) (TVAR(I),I=1,15)
READ (60,50) (VAR(I),I=1,15)
10 FORMAT (8AI0)
20 FORMAT (I5,5X,5E10.3)
30 FORMAT (2A10,2A20,A10)
40 FORMAT (2A10,5E10.3)
50 FORMAT (8E10.3)
```

Table 2 contains input value nomenclature and recommended ranges for the prime cycle independent variables.

**Table 2**  
**Input Data File: CYC\$INP Nomenclature**

Variable Name	Variable Type	Description	Recommended Range
CYCDES(8)	8*CH*10	Title for the series of cases being run.	N/A
IPRINT	INTEG	Key denoting output, if any, desired.	0-3
POWER	REAL	Generator output per PCU, kW	50-5,000
PWRFCTR	REAL	Output power factor	.8-1.0
VOLTAGE	REAL	Line-to-line output voltage, Volts	1,000-10,000
TEMP(9)	REAL	Heat source discharge temperature, K	830-1500
GENASP	REAL	Generator rotor aspect ratio[l/d], dmils	2-3
COMPTYPE	CH*10	Compressor type, 'AXIAL' or 'RADIAL'	
TURBTYP	CH*10	Turbine type, 'AXIAL' or 'RADIAL'	
GENTYPE	CH*20	Generator type, 'RING WOUND TPTL PMG'	
INTTYP	CH*20	Electrical interface type, 'TRANSFORMER'	
CLNTTYPE	CH*10	Coolant description [descriptive only]	'ANY DESC'R
GAS(2)	2*CH*10	Gas(es) used in cycle, options are:	'HELIUM' 'NEON' 'ARGON' 'KRYPTON' 'XENON'
CPCLNT	REAL	Specific heat of coolant, cal/gm-K	.3-7
TINCLNT	REAL	Coolant inlet [turboalt] temp, K	510 max
TOUTCLNT	REAL	Coolant exit [turboalt] temp, K	525 max
XBL	REAL	Working fluid bleed from SP#4 to #3, fract of compressor flowrate, dmils	0-02
XBLT	REAL	Working fluid bleed from SP#4 to #13, fract of compressor flowrate, dmils	0-02
LIFETIME	REAL	Full power lifetime, years	2-10.0
FCTQ8	REAL	Thermal loss fraction of heat into cycle lost from HSA inlet duct	0-03
FCTQ10	REAL	Thermal loss fraction of heat into cycle lost from turbine inlet duct	0-03
FCTQ13	REAL	Thermal loss fraction of heat into cycle lost from turbine discharge duct	0-03
TVAR(14)	14*CH*10	Variable name for subscripted variable:  i = 1 T1 - Compressor inlet temperature, K 2 PRC - Compressor T-T pressure ratio [4'/1'] 3 MW - Working fluid molecular weight, gm/gm-mole 4 ER - Recuperator cold-side temp effectiveness, dmils 5 NSC - Compressor specific speed, rpm- 6 N - rotating speed, rpm 7 DP/P6 - Fractional pressure loss, compressor disch duct 8 DP/PREC - Recuperator total fractional pressure loss, both sides 9 DP/P8 - Fractional pressure loss, HSA inlet duct 10 DP/P9 - Fractional pressure loss, HSA 11 DP/P10 - Fractional pressure loss, turbine inlet duct 12 DP/P13 - Fractional pressure loss, turbine disch duct 13 DP/P15 - Fractional pressure loss, HRA inlet duct 14 DP/P16 - Fractional pressure loss, HRA 15 DP/P17 - Fractional pressure loss, compressor inlet duct	
VAR(15)	REAL	Value of subscripted variable	

A consistent set of statepoint locations was established for use throughout the parametric design code and to facilitate integration of Rocketdyne and NASA support routines. Figure 2 shows a fundamental cycle schematic for a closed Brayton system using an auxiliary cooling loop. A total of seventeen statepoints were needed to adequately characterize the configuration options possible with the code. Some statepoints are relevant to only radial flow turbomachines and their secondary flows and intermediate pressure levels.

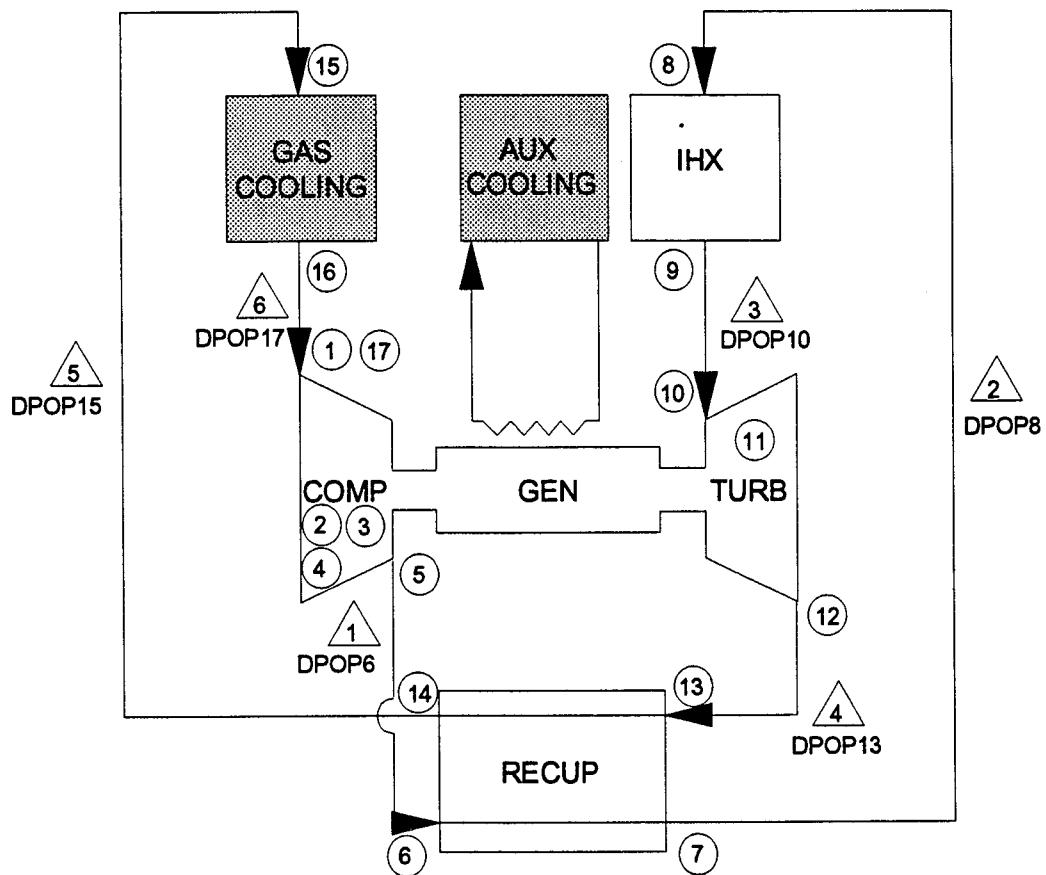
Table 3 also summarizes the statepoint locations with descriptive text. The statepoint location numbers are consistent with all code annotation used throughout the deliverable package.

Table 3. Brayton Statepoint Locations

Statepoint	Location
# 1	- Compressor Inlet
# 2	- Compressor Rotor Tip [static pressure]
# 3	- Compressor Diffuser Inlet
# 4	- Compressor Diffuser Exit [comp-end bleed flow added]
# 5	- Compressor Stage Exit [comp-end bleed flow removed]
# 6	- Recuperator High Pressure Flow Inlet
# 7	- Recuperator High Pressure Flow Discharge
# 8	- HSA Inlet
# 9	- HSA Discharge
# 10	- Turbine Inlet
# 11	- Turbine Nozzle Exit
# 12	- Turbine Discharge
# 13	- Recuperator Low Pressure Flow Inlet [turb-end bleed flow added]
# 14	- Recuperator Low Pressure Flow Discharge
# 15	- HRA Inlet
# 16	- HRA Discharge
# 17	- Compressor Inlet Duct Exit [= to Statepoint #1]

Notes:

- 1) Total [stagnation] conditions except as noted.
- 2) Statepoints # 2, 3, 4 are relevant with radial compressor
- 3) Statepoints # 11 is relevant with radial turbines only



(○) State point no. identification

(△) Duct no. identification

DPOPx Duct pressure drop variable name

Figure 2. Brayton System Statepoint Locations

### 2.2.1.2 Recuperator Input/REC\$INP Input Data File

Subroutine RECINP performs the input function for the recuperator heat transfer matrix definition. RECINP reads data file REC\$INP to establish the physical geometry and properties of the specified heat transfer structure. The routine assumes that a compact rectangular offset fin matrix is used. RECINP calls subroutine RCTFIN [rectangular fin]. RECINP also provides formatted output capability and prints to filename: PROG\$OUT when keyed by variable IPRINT >=1.

Subroutine RCTFIN computes specific geometric parameters related to heat transfer calculations for rectangular plate-fin heat exchanger matrices. Included are parameters such as hydraulic diameters, flow:frontal area ratios, surface areas:unit volume ratios, etc.

The REC\$INP data file establishes the physical geometry and properties of the specified heat transfer structure. An example REC\$INP data file is given in Table 4 showing the structure of the data file and its contents.

Table 4. Example REC\$INP Data File

Plate Fin Recuperator; Kays & London Matrix, Exponent=.615					
.615	.15	.15	.15	.15	
6.3	.0127	.3175	.1524	8.03	22.14
7.87	.0127	.2540	.1206	8.03	22.14
	.0254			8.03	
	.00381			8.03	

The first line in the data file contains the recuperator description. The code will accept up to an 80 character description.

The second line of the data file contains the following information as read from left to right:

- 1) Prandtl number exponent (dimensionless)
- 2) Heat transfer area margin, low pressure side (dimensionless)
- 3) Pressure drop margin, low pressure side (dimensionless)
- 4) Heat transfer area margin, high pressure side (dimensionless)
- 5) Pressure drop margin, high pressure side (dimensionless)

The third line of the data file contains the following information as read from left to right:

- 1) Number of fins/cm, low pressure side ( $\text{cm}^{-1}$ )
- 2) Fin thickness, low pressure side (cm)
- 3) Fin spacing, low pressure side (cm)
- 4) Equivalent fin length, low pressure side (cm)
- 5) Density of fin material, low pressure side (gm/cc)
- 6) Thermal conductivity of fin material, low pressure side (W/m-K)

The fourth line of the data file contains the following information as read from left to right:

- 1) Number of fins/cm, high pressure side ( $\text{cm}^{-1}$ )
- 2) Fin thickness, high pressure side (cm)
- 3) Fin spacing, high pressure side (cm)
- 4) Equivalent fin length, high pressure side (cm)
- 5) Density of fin material, high pressure side (gm/cc)
- 6) Thermal conductivity of fin material, high pressure side (W/m-K)

The fifth line of the data file contains the following information as read from left to right:

- 1) Thickness of the splitter plates (cm)
- 2) Density of the plate material (gm/cc)

The sixth line of the data file contains the following information as read from left to right:

- 1) Thickness of the braze material (cm)
- 2) Density of the braze material (gm/cc)

#### 2.2.1.3 IHX Definition/IHX\$INP Input Data File

The IHX\$INP data file establishes the physical geometry and properties of the specified shell-and-tube heat exchanger system. An example IHX\$INP data file is given in Table 5 showing the structure of the data file and its contents.

Table 5. Example IHX\$INP Data File

1	226.8	1166.7	1111.1	.102	6.0
14.4	.025	8.34	8.34	1.27	

The first line of the data file contains the following information as read from left to right:

- 1) IHXflg - 1=Lithium, 2=NaK
- 2) XUEST1 - Initial value for U overall (W/m<sup>2</sup>-K)
- 3) XTHIN1 - Hot side inlet temperature, K
- 4) XTHOUT1 - Hot side outlet temperature, K
- 5) XTTUBE1 - Tube wall thickness, cm
- 6) ANPLATES1 - Number of support plates

The second line of the data file contains the following information as read from left to right:

- 1) XAKTUBE1 - Tube wall thermal conductivity, W/m-K

- 2) XTINS1 - Insulation thickness, cm  
 3) XDENINS1 - Bulk density of insulation, gm/cc  
 4) XDENSSH1 - Density of shell, gm/cc  
 5) XDTUBE1 - Inside diameter of heat exchanger tubes, cm

#### 2.2.1.4 Ducting Definition/DUCT\$INP Input Data File

The DUCT\$INP data file establishes the physical geometry and properties of the ducting system. An example DUCT\$INP data file is given in Table 6 showing the structure of the data file and its contents. There are six duct segments numbered as previously shown in Figure 2. The data file contains six columns of data, one column for each duct segment. It should be noted that, for a non-recuperated system, the data contained for duct segments 2 and 5 is not used. The data for duct segments 1 and 4 must be adjusted accordingly.

Table 6. Example DUCT\$INP Data File

15.0	15.0	15.0	15.0	15.0	15.0
60.0	60.0	60.0	60.0	60.0	60.0
0.102	0.102	0.102	0.102	0.102	0.102
4.43	8.30	8.30	8.30	4.43	4.43
0.00	100.0	100.0	100.0	0.0	0.0
0.0013	0.0013	0.0013	0.0013	0.0013	0.0013
5.54	5.54	5.54	5.54	5.54	5.54
0.001	0.001	0.001	0.001	0.001	0.001

The data file contains eight rows of data. Row number 1 contains the L/D's of ducting assumed for each duct segment. Row number 2 contains the equivalent L/D's of elbows, etc, assumed for each duct segment. Row number 3 contains the duct wall thickness (cm) assumed for each duct segment. Row number 4 contains the material density (gm/cc) assumed for each duct segment. Row number 5 contains number of layers of multifoil insulation (MFI) assumed for each duct segment. Row number 6 contains the thickness (cm) of the individual MFI layer. Row number 7 contains the material density (gm/cc) assumed for the MFI foils. Row number 8 contains the surface roughness factor assumed for each duct segment.

#### 2.2.1.5 Subroutine BRAYTON

Subroutine BRAYTON is the coordinating "traffic control" code which directs the calling sequence of the support routines. BRAYTON is fundamentally the mainline program for a single PCU design with component sizing. A full nomenclature and statepoint summary is included in the code annotation for this routine.

Subroutine BRAYTON is involved throughout all areas of the code, but performs the following activities associated with data input and setup:

- 1) BRAYTON provides initial values or zeros-out certain variables which are varied during the power output iteration.
- 2) BRAYTON calls Subroutine SETUP which loads and aligns variables from the input array
- 3) BRAYTON calls Subroutine GASPROP once per case to establish the constituency of the working fluid and the transport parameters to be used for property calculations.

#### 2.2.1.6 Subroutine SETUP

Subroutine SETUP exists as a convenience for managing cycle design variables which would commonly be involved in a parametric optimization of a Brayton system. As delivered, fourteen such parameters are described by the array value in VAR(i). The companion variable names are contained in the array TVAR(i).

VAR(i) and TVAR(i) are read, or otherwise defined, by the mainline program. Currently these values are read in by MANE from input filename: CYC\$INP. Subroutine SETUP simply loads the value contained in VAR(i) into the appropriate [recognizable] cycle variable. Subroutine SETUP also sets and/or initializes certain system pressure loss functions as required by the recuperation scheme. When VAR(4) [recuperator effectiveness]  $\leq 0$  the system is treated as non-recuperated which means that two of the six ducting pressure losses do not exist. When VAR(4)  $> 0$  the recuperator pressure loss is divided by approximation between the two flow streams.

#### 2.2.1.7 Subroutine GASPROP

Subroutine GASPROP [gas properties] is called at the start of a new case to establish the constituent gases and mixture to be utilized for the cycle working fluid. The gas constituents are defined in filename: CYC\$INP by variables GAS(2). The molecular weight is defined by the value of VAR(3). GASPROP establishes the composition required to provide the specified molecular weight, the fluid specific heat [ $C_p$ ], and the Lennard-Jones collision parameters to support calculation of transport properties during the analysis. The collision integrals to support subsequent calculations are loaded from DATA statements in GASPROP.

## 2.2.2 Thermodynamic Model

Subroutine BRAYTON is the coordinating "traffic control" code which directs the calling sequence of the support routines. BRAYTON is fundamentally the mainline program for a single PCU design with component sizing. A full nomenclature and statepoint summary is included in the code annotation for this routine.

BRAYTON calls Subroutine STATEPT which accomplishes the thermodynamic and most of the geometric definition of the conversion system.

### 2.2.2.1 Subroutine STATEPT

Subroutine STATEPT performs basic thermodynamic calculations and sequences the use of component design routines. The functions accomplished by STATEPT [roughly in sequence presented] are:

- Computes component pressure ratios
- Call Subroutine GENSIZE
- Initializes Comp Disch condns. based on first approx
- Checks compressor specific speed value versus boundary
- Initialize all flows based on compressor inlet flow
- Initialize all system pressures based on compressor inlet pressure
- Compute thermal losses
- Compute all statepoint temperatures and enthalpies
- Call GENSIZE get sizes, cooling load
- Call BEARING define shaft power loss
- Call WINDLOS define shaft power loss
- Call AXCOMP or RADCOMP, define compressor geometry and performance
- Call AXTURB or RADTURB, define turbine geometry and performance
- Check convergence on power, scale system pressures if required

### 2.2.2.2 Subroutine BEARING

This routine estimates the parasitic shaft power loss associated with the rotor suspension [bearing] subsystem. The routine estimates bearing loss equal to 2 % of the gross generator terminal power. This value has proven to be reasonable in previous scoping studies in the higher power ranges.

### 2.2.2.3 Subroutine GENSIZE

Subroutine GENSIZE provides accurate preliminary sizing and performance for a ring-wound two-pole toothless [RW TPTL] permanent magnet generator [PMG]. Generator mass and several key dimensions are estimated. This routine is identical to the routine developed to support potassium Rankine systems. The details of the generator algorithm development are provided in Section 2.2.4 and Appendix B.

GENSIZE requires that rotating speed, power level, power factor, voltage, and rotor aspect ratio be furnished. All designs are limited to a maximum 700 ft/sec

rotor surface speed based on stress and magnetic design considerations. If the input results in surface speed in excess of 700 ft/sec, GENSIZE resets the rotating speed to a value which permits a design to be completed.

GENSIZE will generate warning or error messages if out-of-limits operation is requested. GENSIZE source code is liberally annotated explaining all assumptions.

#### 2.2.2.4 Subroutine WINDLOS

Subroutine WINDLOS computes the parasitic shaft power loss predicted to result from the current generator design and operating conditions. The calculations are based on the methods presented in NASA TM X-67809 for rotating cylinders within a relatively smooth-bore cylinder. The constant and Reynolds No. exponent used in WINDAGE are fitted from Figure 6 in the reference TM X.

#### 2.2.2.5 Compressor Parametric Design Routines

Routines to support design of both axial and radial compressors are provided. The routine used is determined by the input variable COMPTYPE read from CYC\$INP which denotes either "AXIAL" or "RADIAL".

Subroutine AXCOMP - Subroutine AXCOMP serves to provide sizing and performance estimates for a selected multi-stage, subsonic axial compressor. AXCOMP defines the inner and outer flowpath dimensions at the compressor inlet and outlet planes. The hub radius is assumed constant.

The code operates in a manner to find the compressor design with the fewest number of stages which does not violate the geometric or Mach Number limits imposed. The specific limits imposed are:

- a) hub/tip radius ratios must be  $\geq .60$
- b) hub/tip radius ratios must be  $\leq .95$
- c) relative Mach No. must be  $\leq .80$

If a solution is not found with 15 or fewer stages, an error message is generated and the adiabatic efficiency is set to 60 %.

AXCOMP estimates the adiabatic efficiency of its design based a polytropic efficiency algorithm, a size correction, and a tip clearance loss correlation. Figure 3 shows Balje's correlation total-to-static polytropic efficiency for axial compressors versus specific speed. This efficiency correlation is for compressors with 2% [of blade height] clearance ratio and a Reynolds Number of 2 million. Figure 4 provides an efficiency multiplier which corrects for clearance ratios other than 2%. Figure 5 provides a Reynolds Number correction loss multiplier suitable for both radial and axial compressors.

AXCOMP also computes the value of  $AN^2$  [Flow Area (sq in) x Speed [rpm]<sup>2</sup>] which is used to evaluate the centrifugal stress levels expected in the candidate

design. Maximum blade root stress is estimated using an algorithm. A limit of 30,000 psi mean root stress is assumed. Material density is consistent with Titanium alloys. The value of AN<sup>2</sup> is written to file: PROG\$OUT if keyed by IPRINT.

Subroutine RADCOMP - Subroutine RADCOMP serves to provide sizing and performance estimates for a selected radial compressor design.

RADCOMP estimates the adiabatic efficiency of its design based a polytropic efficiency versus specific speed algorithm, a Reynolds Number correction, a physical size correction, and a shroud clearance loss correlation. Figure 6 show Balje's correlation total-to-static polytropic efficiency for radial compressors versus specific speed. This efficiency correlation is for compressors with 2% [of "B" width] clearance ratio and a Reynolds Number of 2 million. Figure 7 provides an efficiency multiplier which corrects for clearance ratios other than 2%. Figure 5 is also used for radial compressors and provides a Reynolds Number correction loss multiplier suitable for both radial and axial compressors. The rotor tip speed is calculated and compared to limit values. Tip speed is also available to output through filename PROG\$OUT.

#### 2.2.2.6 Turbine Parametric Design Routines

Routines to support design of both axial and radial turbine are provided. The routine used is determined by the input variable TURBTYPE read form CYC\$INP which denotes either "AXIAL'or'RADIAL'.

Subroutines AXTURB & STRNGTH - Subroutine AXTURB serves to provide sizing and performance estimates for a selected multi-stage axial turbine. AXTURB defines the inner and outer flowpath dimensions at the turbine inlet and outlet planes. The hub radius is assumed constant.

The code operates in a manner to find the turbine design with stage specific speed closest to .55 [dmls] which does not violate the geometric limits imposed. The specific limits imposed are:

- a) hub/tip radius ratios must be  $\geq .60$
- b) hub/tip radius ratios must be  $\leq .90$

Figure 8 shows Balje's correlation total-to-static polytropic efficiency for axial turbines versus specific speed. This efficiency correlation is for turbines with 2% [of blade height] clearance ratio and a Reynolds Number of 2 million. Figure 9 provides an efficiency multiplier which corrects for clearance ratios other than 2%. Figure 10 provides a Reynolds Number correction loss multiplier suitable for both radial and axial turbines.

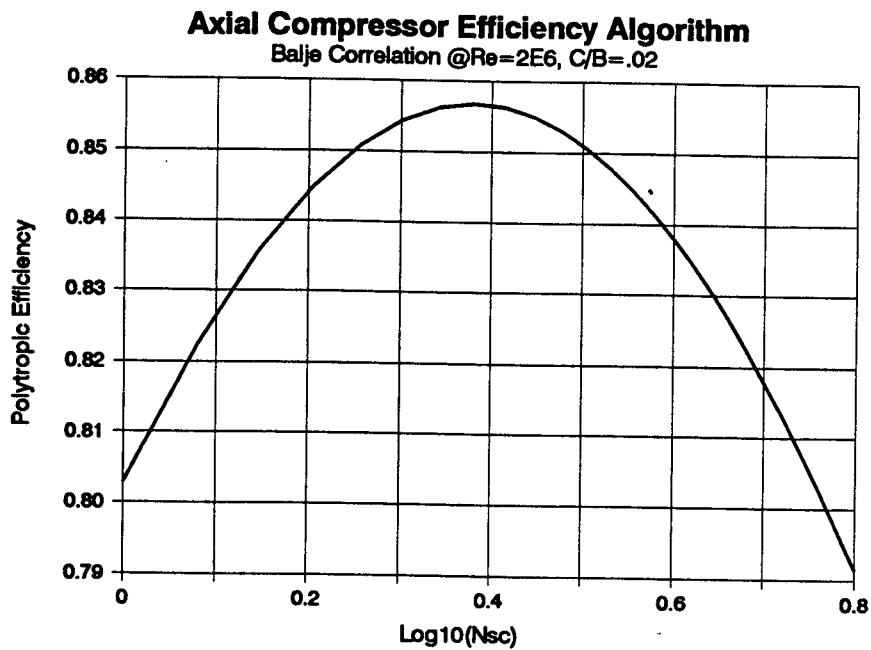


Figure 3. Axial Compressor Total-to-Static Polytropic Efficiency versus Specific Speed

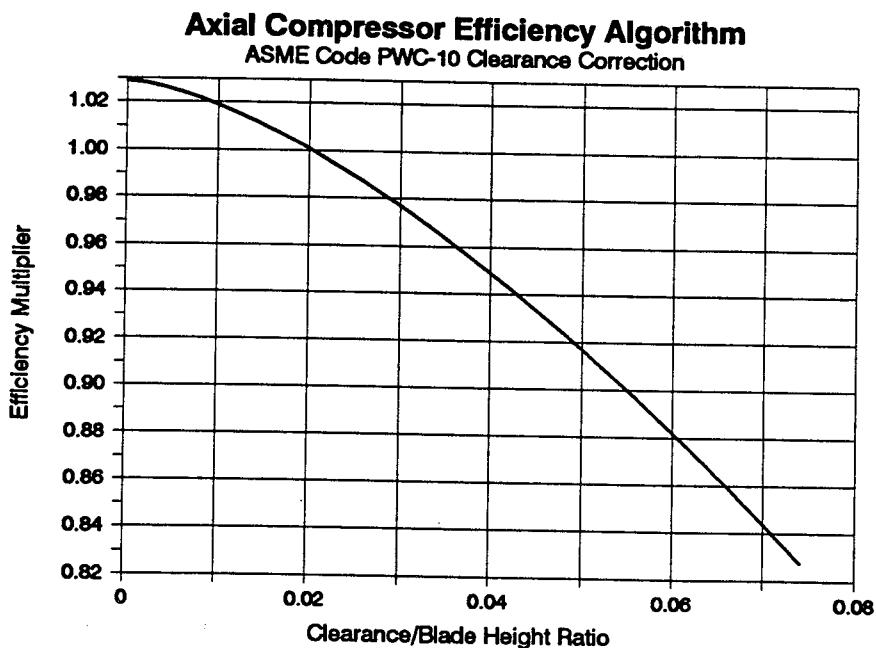


Figure 4. Axial Compressor Tip Clearance Efficiency Correction

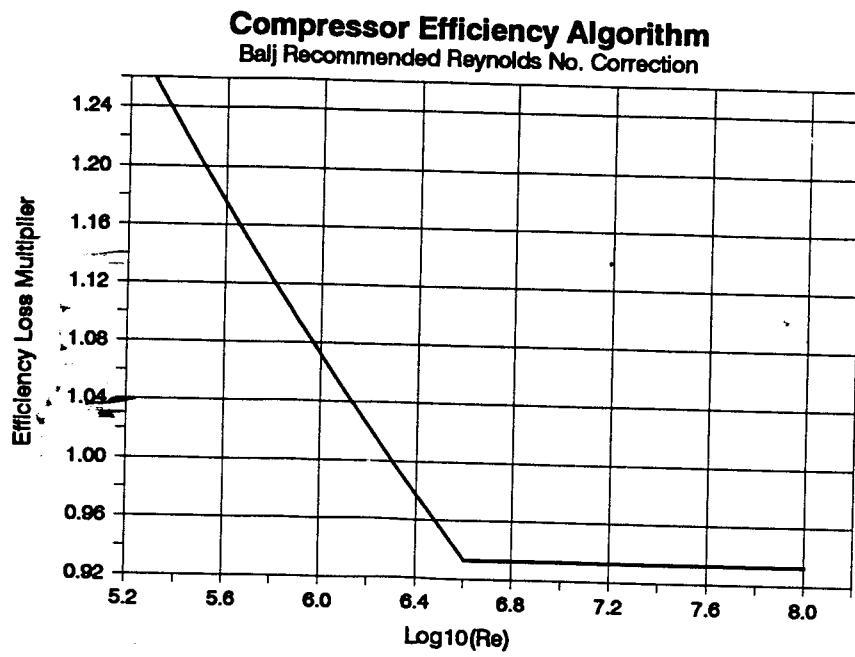


Figure 5. Axial and Radial Compressor Reynolds Number Efficiency Correction

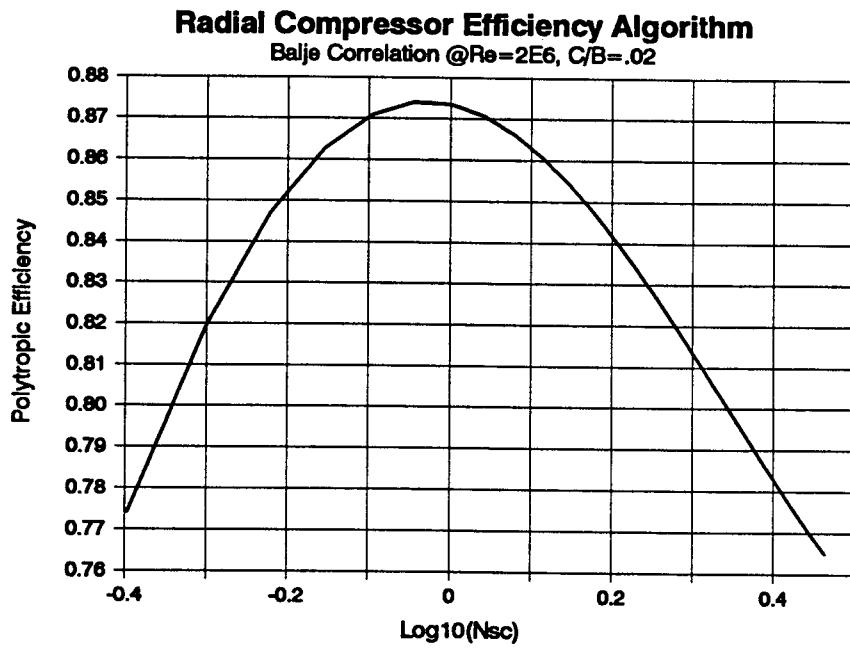


Figure 6. Radial Compressor Total-to-Static Polytropic Efficiency versus Specific Speed

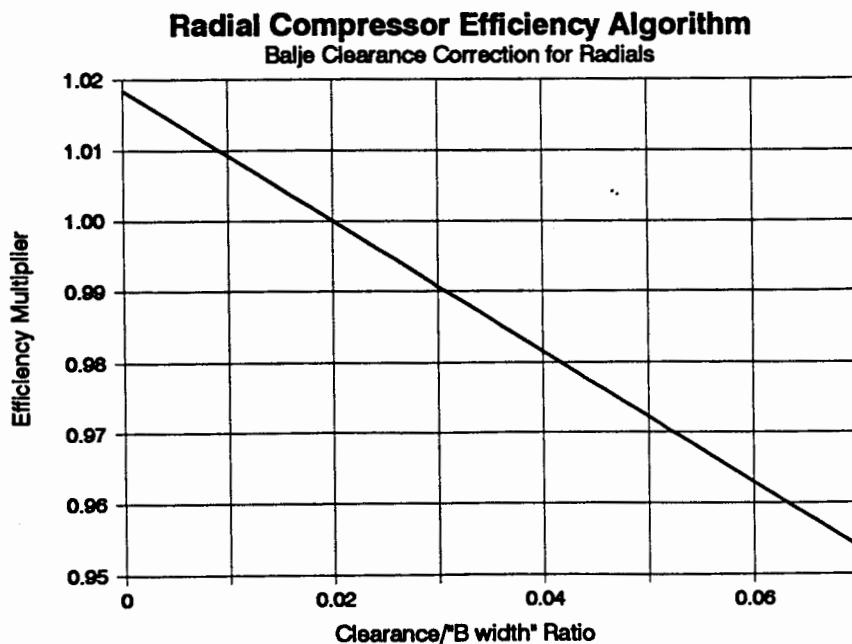


Figure 7. Radial Compressor Tip Clearance Efficiency Correction

AXTURB estimates the adiabatic efficiency of its design based a polytropic efficiency algorithm, a size correction, and a tip clearance loss correlation. AXTURB also computes the value of  $AN^2$  [Flow Area (sq in)  $\times$  Speed (rpm) $^2$ ] which is used to evaluate the centrifugal stress levels expected in the candidate design. Subroutine AXTURB uses Subroutine STRNGTH to ascertain the allowable blade root stress level to meet the required life at the specified operating temperature. Subroutine STRNGTH is currently using a 1% creep criteria [-3 sigma properties] for TZM refractory alloy.

The value of  $AN^2$  is written to file: PROG\$OUT if keyed by IPRINT. The stress limited  $AN^2$  is also printed in the same output file.

Subroutine RADTURB - Subroutine RADTURB serves to provide sizing and performance estimates for a radial turbine defined by virtue of specified operating conditions. RADTURB defines the rotor tip and exducer hub and tip radii.

Figure 11 show Balje's correlation total-to-static polytropic efficiency for radial compressors versus specific speed. This efficiency correlation is for compressors with 2% [of blade height] clearance ratio and a Reynolds Number of 2 million. Figure 12 provides an efficiency multiplier which corrects for clearance ratios other than 2%. Figure 10 provides a Reynolds Number correction loss multiplier suitable for both radial and axial compressors.

RADTURB defines the rotor tip speed and radius which correspond to the maximum efficiency ratio to the ideal spouting velocity. The exducer shroud radius is constrained to be between .70 and .90 times the tip radius. Exducer hub radius must be .40 or more times the exducer tip radius.

#### 2.2.2.7 Gas Property Support Functions

Functions VISCOS and CONDUCT use the Lennard-Jones parameters established by GASPROP to compute working fluid viscosity and thermal conductivity as a function of the specified temperature level. Functions AVVISC and AVCOND generate an estimate of the effective average viscosity or conductivity over a specified range of temperature using Simpson's Rule.

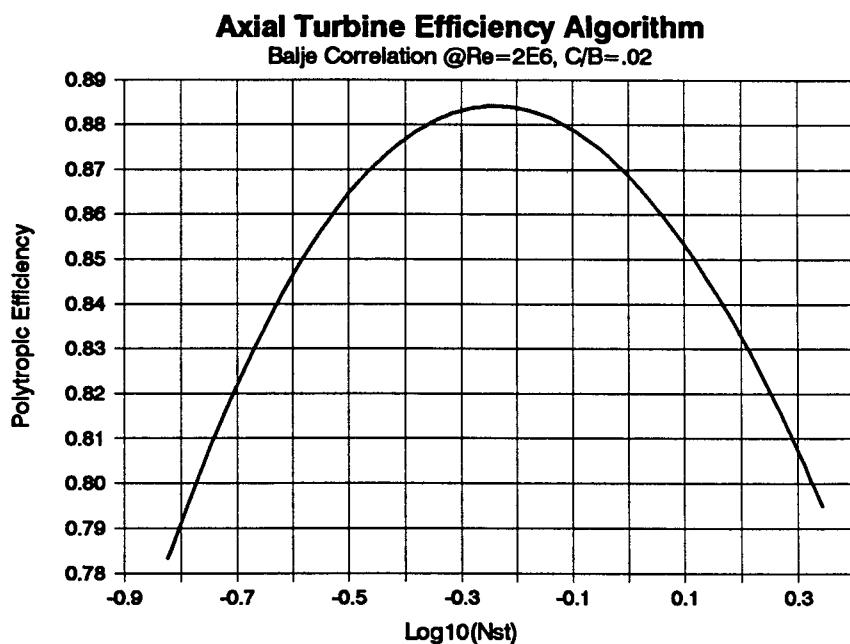


Figure 8. Axial Turbine Total-to-Static Polytropic Efficiency versus Specific Speed

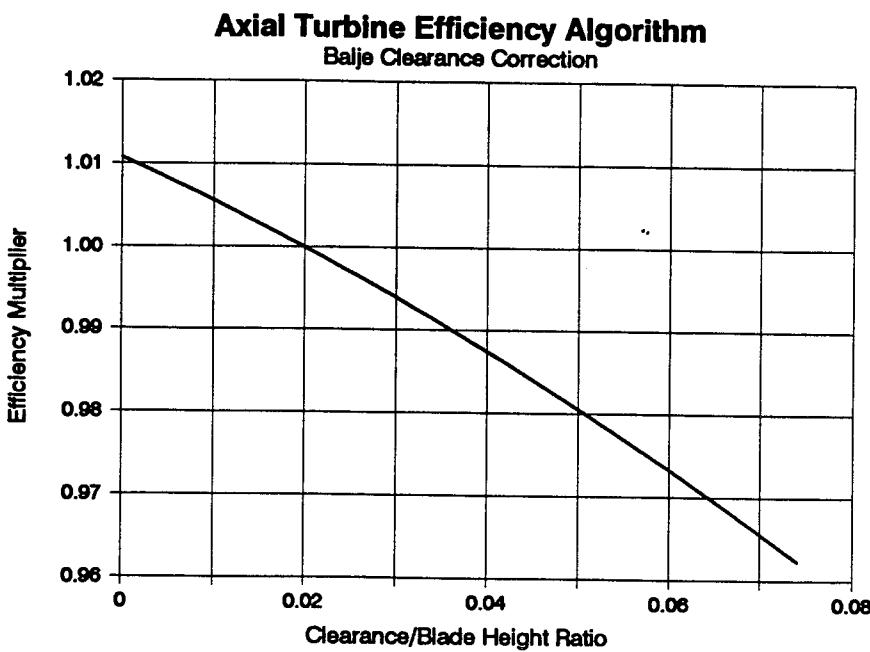


Figure 9. Axial Turbine Tip Clearance Efficiency Correction

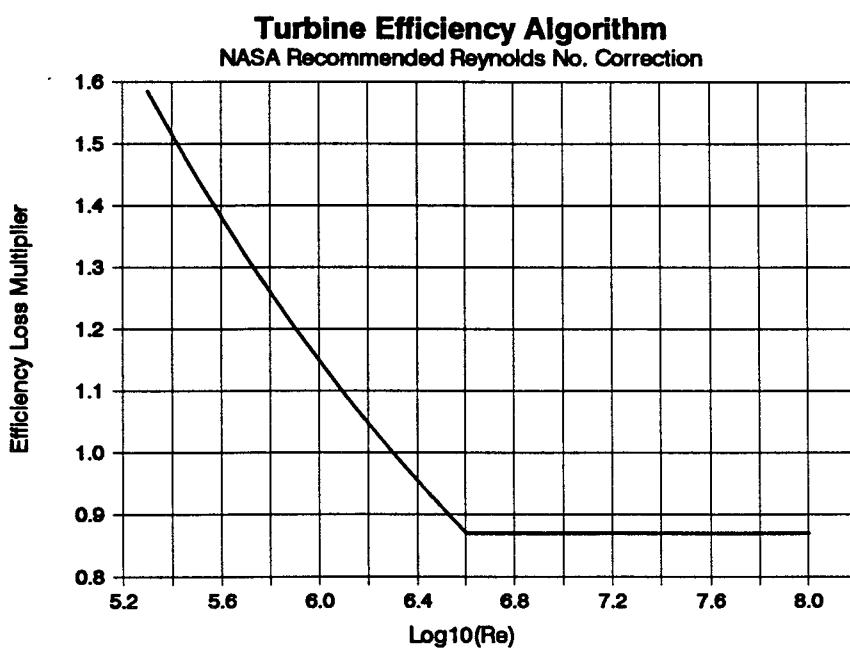


Figure 10 Axial and Radial Turbine Reynolds Number Efficiency Correction

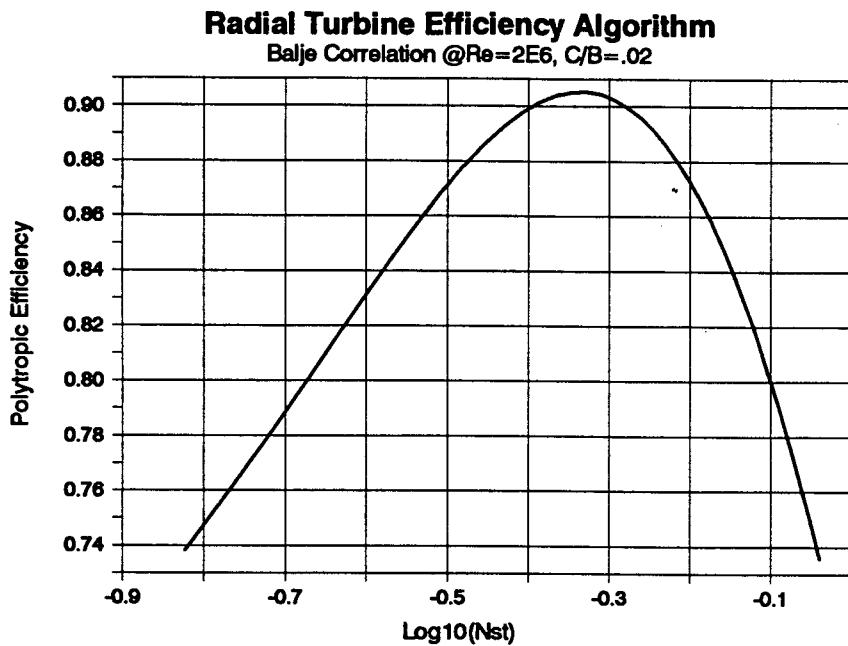


Figure 11 Radial Turbine Total-to-Static Polytropic Efficiency versus Specific Speed

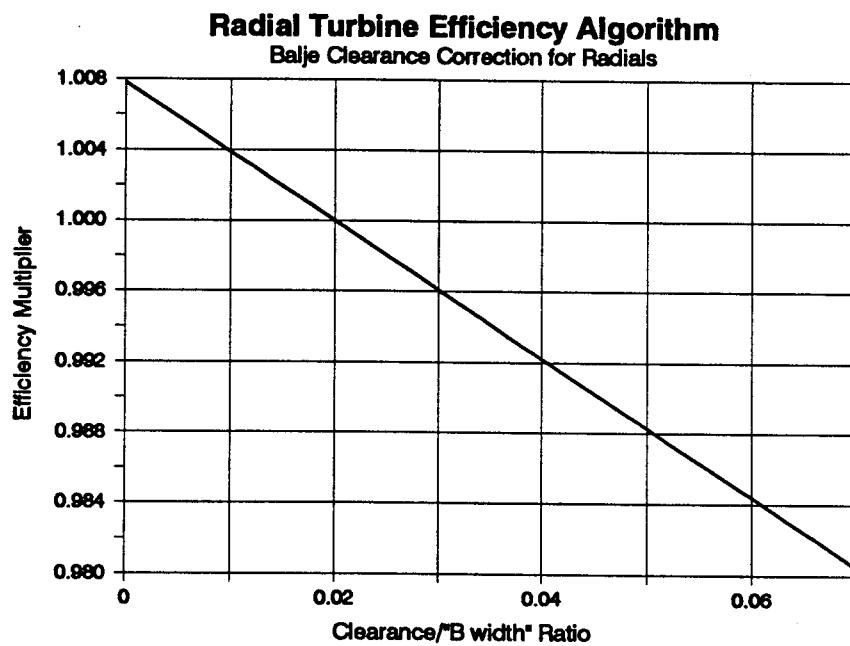


Figure 12 Radial Turbine Tip Clearance Efficiency Correction

## **2.2.3 Postprocessor/Output Module**

### **2.2.3.1 Subroutine BRAYTON**

Subroutine BRAYTON is the coordinating "traffic control" code which directs the calling sequence of the support routines. BRAYTON is fundamentally the mainline program for a single PCU design with component sizing. A full nomenclature and statepoint summary is included in the code annotation for this routine.

Subroutine BRAYTON performs the following functions in the postprocessor/algorithm section:

- Calls DUCTING, defines dimensions/mass of each duct segment.
- Calls IHXSHEL, provides for gas-side pressure drop iteration to complete IHX sizing.
- Provides for an iteration (to be set-up by the NASA integrator of the code) when calling the heat rejection module in order to complete sizing of the heat rejection heat exchanger within the desired pressure drop.
- Calls RECSIZE, defines dimension/mass
- Calls TAC, defines compressor and turbine masses and sums these with the generator mass and structural mass estimate to predict complete TAC mass
- If IPRINT > 0 BRAYTON writes a cycle summary page to PROG\$OUT and returns to calling routine

### **2.2.3.2 Subroutine DUCTING**

Subroutine DUCTING sizes four to six duct segments [depending on the recuperation scheme] based on the input data provided to the program. The ducting routine provides diameter, length, and mass information for inclusion in the system design summary. The ducting routine is iterative, solving for a duct diameter (and corresponding length based on the input L/D ratio) for each duct segment satisfying the pressure drop requirement for that duct segment. Warning messages have been provided for designs resulting in low or high gas velocities. The user should evaluate whether the gas velocity makes sense for the case being examined. If not, adjustments should be made to the input data files to change the pressure drop allocation for the duct segment and/or the number of L/D's specified for that duct segment.

### **2.2.3.3 Subroutine IHXSHEL**

Subroutine IHXSHEL sizes and computes the thermal hydraulic performance of the shell-and-tube intermediate heat exchanger. The IHXSHEL subroutine uses the same routine as used in the heat rejection module for shell-and-tube heat exchanger sizing. The routine assumes an system configuration using liquid metal on the tube-side of the heat exchanger and the Brayton cycle gas on the shell-side. The IHXSHEL routine is iterative, solving for a tube pitch ratio satisfying the gas-side pressure drop requirement for the IHX. The user should evaluate whether the tube pitch ratio makes sense for the case being examined. If not, adjustments should be made to the input data file to change the pressure drop allocation for the IHX. Details of the of the shell-and-tube heat exchanger sizing code are

provided in the documentation for the heat rejection module, NASA CR-191132.

#### **2.2.3.4 Provisions for Calling the Heat Rejection Module**

Provisions are provided for establishing an iteration scheme to size the heat rejection heat exchanger to meet the required gas-side pressure drop. The heat rejection heat exchanger sizing routine is part to the Heat Rejection Module supplied with Task Order 19. The NASA integrator will be required to integrate these two codes into the overall system code.

#### **2.2.3.5 Subroutine TAC**

Subroutine TAC is the routine which defines the turbogenerator-compressor (aka turboalternator-...) mass as defined by a system of algorithms. The generator mass is provided by GENSIZE. The development of the GENSIZE algorithms is discussed in section 2.2.4. Compressor and turbine masses (for either axial or radial equipment) are computed as functions of the physical parameters and dimensions defined in the aerodynamic routines.

Mass estimates for all aerodynamic components are base on algorithms derived from gas turbine and closed-cycle data bases. All compressor mass estimates assume full Titanium alloy construction. All turbine mass estimates assume TZM rotating groups [wheels and blades] and a Nb alloy static structure.

#### **2.2.3.6 Subroutine RECSIZE**

Subroutine RECSIZE provides an estimate of recuperator core dimension and mass based on the current cycle statepoint values, the core matrix geometry provided via Subroutine RECINP, and the gas transport properties calculated for the gas or gas mixture specified. Since the integration and packaging of the recuperator is an open design issue, the mass associated with wrap-up, end-sections and manifolds is handled as an added 50% mass over and above the computed core mass. An allocation of 30% of the specified fractional pressure loss is also made for the recuperator entry and exit transition [e.g.; the core pressure loss used for sizing is 70% of the total allocation].

The sizing of the recuperator core is done using friction factor [ $f$ ] and heat transfer correlations [Colburn j-factor] for a typical core matrix. The  $f$  &  $j$  data used were obtained from "Compact Heat Exchanger" by Kays and London and are represented in Figure 13. This data is deemed typical of likely plate-fin constructions using offset fin geometries. Since the data is correlated versus Reynolds Number, it is satisfactory for use in scoping analyses with any geometry similar to the reference design.

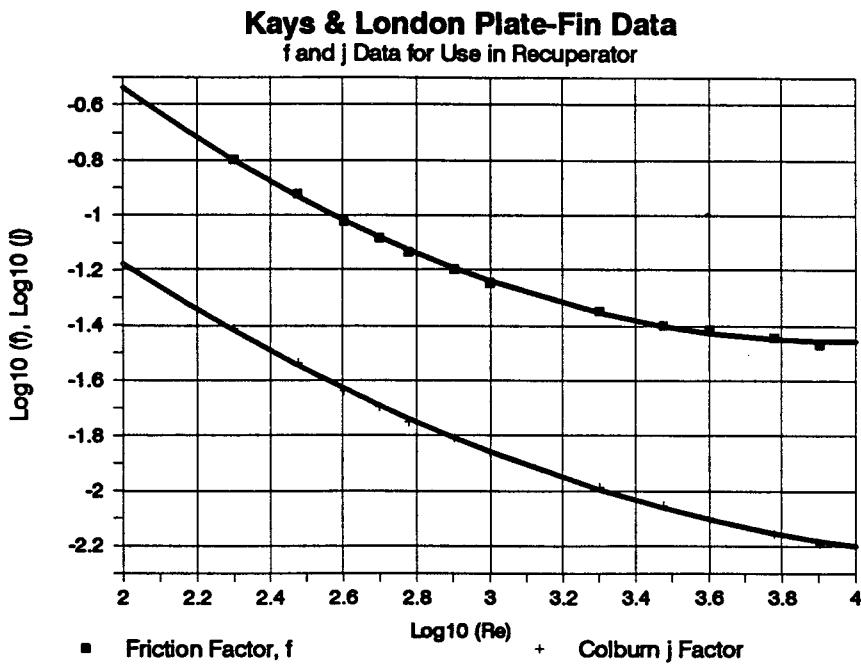


Figure 13 Recuperator Friction Factor and Heat Transfer Correlations

#### 2.2.3.7 Subroutine OUTPUT

Subroutine OUTPUT will provide a one-page cycle/system summary written to filename: PROG\$OUT when the print flag "IPRINT" is > 0. Included in the file are:

- 1) a recap of the independent variables
- 2) a mass breakdown by major component
- 3) aero component geometries and key indicator parameters
- 4) descriptive summary of the resulting generator design
- 5) a full statepoint summary of flows, temps, pressures
- 6) a summary of predicted shaft and thermal parasitic losses

## **2.2.4 Development of Generator Algorithms**

This section discusses the development of the generator design algorithm, GENSIZE, in support of the power conversion systems code development. Specifically, numerous point design studies have been completed and algorithms developed to support generator sizing in the full-up system evaluation code.

### **2.2.4.1 Study Guidelines**

#### Generator Type

All generator designs studied are high performance, high reliability TPTL [two-pole toothless] PM [permanent magnet] type. Both ring wound [RW]/variable cross-section conductor [VCSC] and conventionally wound TPTL configurations were investigated. Specific operating requirements imposed are summarized in Table 7, below.

#### Operating Speed

The TPTL machines were designed to achieve maximum rotor speed consistent with high-reliability (.99+) and 2 to 10 year life. Some advances beyond the state-of-the-art could reasonably be assumed since the use dates range from 2000 to 2015. Although the determination of design speed for a turbo-generator is probably dictated by the generator, the generator speed was also limited to the maximum turbine design speed profile shown in Table 7. No overspeed allowance was included.

#### Operating Environment

The generator designs are primarily intended for use in a Potassium turbo-generator power system. The rotor/wire-support gap is assumed filled with Potassium vapor at the conditions listed in Table 7.

#### Insulation System

A 220 deg C insulation system was selected as the reference system. Some deration of the operating temperature may be required to achieve the more ambitious reliability and life goals. Generator sizing, however, can be accomplished at the nominal 220 deg C for hot-spot temperature. Insulation thickness was based on a potential of 50 volts/mil.

#### Cooling

The generator cooling assumes direct stator cooling with an organic coolant (e.g. n-Heptane, Dowtherm, etc.).

#### Design Voltage

The generator designs considered produce 3-phase alternating current at an RMS line-to-line voltage of either 1400 or 8000. The relationship of desired voltage to generator power level is shown in Table 7.

## Generator Interfaces

The generator designs evaluated were optimized for an assumed transformer interface. The projected power factor for all cases is 0.90 lagging. This interface is more likely than a rectifier interface for an NEP application. The power factors for use in design are included in Table 7.

### Mass/Performance Trade

Overall generator conversion efficiency (including windage) is the salient parameter affecting system optimization. The TPTL designs were optimized to maximum efficiency with a mass/efficiency trade ratio of approximately .2 pounds/kWe generator mass/% generator efficiency.

Table 7  
Generator Design Requirements

Generator Power Output -kVA-	56	111	222	555	1,111	2,222	5,555
Generator type	<----- TPTL PM ----->						
Maximum Speed, krpm	160	120	85	57	41	30	20
Voltage (RMSv 1-1)	1,400	1,400	1,400	1,400	8,000	8,000	8,000
Power Factor	<----- 0.90 lagging ----->						
Gap Conditions:							
Viscos., lb/ft-hr	<----- 0.030 ----->						
Temp, deg F	<----- 440 ----->						
Press, psia	<----- 4.88E-4 ----->						
Rho, lb/cu ft	<----- 1.97E-6 ----->						
Voltage Regulation	<----- +/- 10% ----->						
Insulation Class	<----- 220 C ----->						
Rotor Magnet L/D	<----- <=2.5 ----->						

### 2.2.4.2 Generator Design Results

A total of twenty-one point designs were completed using an AiResearch Los Angeles Division [ALAD] proprietary design code. From the results of these studies the VCSC-RW PMG configuration was selected for inclusion in the deliverable generator sizing code. The point designs were reduced to algorithm form to predict performance, mass, and size as a function of design kVA, rotor

surface speed and desired output voltage.

#### Assumptions, Justification, Limits

- a) A maximum allowable generator rotor surface speed of 700 ft/sec was established by ALAD. Above this speed, the primary flux gap widens rapidly due to the hoop thickness required to retain the rotor magnet.
- b) A reference rotor L/D of 2.5 was selected for the study. The algorithms developed are assumed valid in the range of  $2 \leq L/D \leq 3$  when corrected for L/D not equal 2.5.
- c) The algorithms are assumed valid in the range of output voltage from 1 to 10 kV 1-1, RMS and the range of power factor from .70 to 1.0.
- d) The design analyses were completed assuming 500 deg F operating temperature for both rotor and stator. These assumptions effect magnet aging design margin, electrical insulation life, and conductor resistivity.
- e) The alternator will be integrated for use with direct stator cooling using an organic coolant such a Dowtherm A, N-Heptane, etc.

#### Point Design Study Results

A total of twenty-one point designs were completed using an AiResearch Los Angeles Division [ALAD] proprietary design code. This code provides a comprehensive physical and performance description of candidate generator designs.

The cases run represent three separate data sets run at the power levels defined in Table 7 and the configuration below:

Set A - Conventionally Wound TPTL PMG at 700 ft/sec surface speed

Set B - Ring Wound/Variable Cross-section Conductor TPTL PMG at 700 ft/sec surface speed

Set C - Ring Wound/Variable Cross-section Conductor TPTL PMG at 500 ft/sec surface speed

Data sets A and B were run concurrently with common groundrules to establish the preferred configuration [ring wound or conventional] for continued study.

Tables 8 and 9 summarize the geometries and performance which resulted from the comparison. It can readily be seen that the VCSC-RW TPTL PM machine is the preferred choice for all power ratings studied. The higher efficiency, lower mass, and higher operating speed are made possible by the higher machine air gap flux density resulting from the VCSC-RW design. In addition, better winding space utilization and higher reliability are achieved since the concentrated individual phase windings are located in physically separate 60 degree phase sectors. The borders of these phase sectors are insulated phase-to-phase, while within the sector only turn-to-turn and winding-to-ground insulation is required.

In contrast, stator windings using conventional slotted configurations use two coil sides per slot. These coil sides are associated with different phase windings. Full phase-to-phase voltage potential exists between the coil sides as well as between the phase windings which cross over each other in the end turns. Even though fully insulated, areas of phase windings in contact still exist. This condition limits stator robustness, particularly in severe environments and high voltage designs, and reduces stator reliability.

The ring wound stator configuration that uses single-layer variable cross-section conductors readily lends itself the optimization of the winding to achieve high machine air gap flux density, efficient cooling, and maximum reliability. The most valuable space for an electrical machine [motor or generator] is the area between the surface of the rotor magnet and the ID of the laminated iron flux return path. The smallest possible distance between them yields the highest air gap flux density which leads to the smallest machine mass and size. For the ring wound configuration, the space around the ends of the OD of the flux collector ring is available for much larger conductor segments. Using a high current density Litz wire conductor in the air gap area that is connected to a much larger conductor used for the remainder of the winding results in an enhanced electromagnetic and thermal design. The large cross-section, low current density conductor segment can provide a heat sink and more thermal mass for the winding and thus more effective cooling of the higher current density Litz conductor segment. Lower total winding resistance will result in lower  $I^2R$  losses and higher efficiency.

Table 10 contains design specifics for a series of VCSC-RW TPTL designs operating at 500 ft/sec surface speed. The units are surprisingly low in mass and exhibit small rotor sizes as well. This excellent result at 500 ft/sec is attributed to the much reduced thickness required for the magnet retaining hoop and the resulting large increase in gap flux density. In most cases, rotor sizes are comparable to their 700 ft/sec counterparts and total masses are generally lower.

Table 8 Design Summary for Conventionally Wound TPTL Generators Operating at 700 ft/sec Surface Speed

		Power Level						
Power	kWe	50	100	200	500	1,000	2,000	5,000
kVA	kVA	56	111	222	556	1,111	2,222	5,556
Alt Type	TPTL	Conv.	Conv.	Conv.	Conv.	Conv.	Conv.	Conv.
N	rpm	54,500	48,200	34,200	25,000	18,400	13,700	10,240
Vbase	l-n RMS	808	808	808	808	4,620	4,620	4,620
Rotor Dia	inches	2.94	3.55	4.69	6.42	8.72	11.71	15.67
Stator OD	inches	5.75	6.52	8.53	11.05	13.51	17.70	22.70
Total Len	inches	12.37	14.20	18.60	24.30	33.60	44.30	59.50
Magnet L/D	dms	2.57	2.51	2.5	2.45	2.53	2.5	2.58
Xcom	P.U.	0.121	0.119	0.129	0.114	0.131	0.130	0.137
EM Mass	lbm	38.48	65.03	146.30	358.2	724.0	1682.0	4101.0
Rotor Mass	lbm	15.58	26.52	61.00	152.0	395.0	940.0	2314.0
Total Effy	%	95.08%	95.88%	96.14%	96.65%	95.40%	95.80%	96.35%
Total Loss	kW	2.56	4.30	8.03	17.3	48.3	88.2	189.3
Rotor Tip Spd	ft/s	700	700	700	700	700	700	700
Bcore	kL/sq in	80	80	80	80	140	140	140

Table 9 Design Summary for Ring-Wound TPTL Generators Operating at 700 ft/sec Surface Speed

		Power Level						
Power	kWe	50	100	200	500	1,000	2,000	5,000
kVA	kVA	56	111	222	556	1,111	2,222	5,556
Alt Type	TPTL	R.W.	R.W.	R.W.	R.W.	R.W.	R.W.	R.W.
N	rpm	80,000	62,000	47,500	32,500	23,000	16,500	11,400
Vbase	l-n RMS	808	808	808	808	4,620	4,620	4,620
Rotor Dia	inches	2.01	2.59	3.38	4.92	6.98	9.72	14.07
Stator OD	inches	3.82	4.80	6.10	8.60	10.80	14.50	20.00
Total Len	inches	6.20	7.40	9.80	14.30	19.00	27.20	38.00
Magnet L/D	dms	2.49	2.49	2.5	2.5	2.49	2.5	2.51
Xcom	P.U.	0.120	0.120	0.130	0.130	0.130	0.120	0.130
EM Mass	lbm	14.08	29.40	62.00	182.7	375.0	993.0	3105.0
Rotor Mass	lbm	4.82	10.32	23.00	71.0	198.0	536.0	1630.0
Total Effy	%	96.39%	96.68%	96.80%	96.97%	96.44%	96.30%	96.97%
Total Loss	kW	1.86	3.44	6.60	15.6	36.9	77.0	156.0
Rotor Tip Spd	ft/s	700	700	700	700	700	700	700
Bcore	kL/sq in	80	80	80	80	140	140	140

Table 10 Design Summary for Ring-Wound TPTL Generators Operating at 500 ft/sec Surface Speed

Power	kWe	Power Level						
		50	100	200	500	1,000	2,000	5,000
kVA	kVA	56	111	222	556	1,111	2,222	5,556
Alt Type	TPTL	R.W.	R.W.	R.W.	R.W.	R.W.	R.W.	R.W.
N	rpm	61,200	47,300	36,800	25,700	18,000	14,050	9,700
Vbase	l-n RMS	808	808	808	808	4,620	4,620	4,620
Rotor Dia	inches	1.87	2.42	3.11	4.46	6.37	8.16	11.81
Stator OD	inches	4.00	5.60	6.50	9.10	10.80	13.30	18.70
Total Len	inches	5.60	7.20	9.00	13.00	17.70	22.70	32.80
Magnet L/D	cm/l	2.51	2.50	2.50	2.50	2.50	2.50	2.50
Xcom	P.U.	0.130	0.130	0.130	0.130	0.120	0.130	0.130
EM Mass	lbm	14.56	29.63	60.12	168.0	348.9	729.7	2131.0
Rotor Mass	lbm	3.89	8.40	17.70	51.9	152.0	316.0	960.0
Total Effy	%	96.55%	96.71%	96.76%	96.88%	96.29%	96.49%	96.69%
Total Loss	kW	1.77	3.40	6.69	16.1	38.6	72.7	171.4
Rotor Tip Spd	ft/s	500	500	500	500	500	500	500
Bcore	kL/sq in	80	80	80	80	140	140	140

Table 11. Generator Materials and Technology Assumptions

Component	Material	Salient Info	Technology Status
Rotor Magnet	Samarium-Cobalt	30 MGO	Commer. Avail., Select Mat'l
Rotor Hoop	Inconel	180 ksi	Commer. Avail., Special Order
Outer Condctrs	Copper		Commer. Avail., Mil-W
Inner Condctrs	Litz Wire		Comm. Available
Stator Insultn	Pyre-ML	Organic	Comm. Available
Flux Ret. Path	Si-steel [3.5%] 50-750 kWe	80 kL/in <sup>2</sup>	Comm. Available
Flux Ret. Path	Hyperco 750-5000 kWe	140 kL/in <sup>2</sup>	Comm. Available
Support Struct	Polyamide		Comm. Available

### Materials of Construction, Technology Assumptions

Table 11 contains a summary of the materials of construction assumed in the point design study and performance algorithm. Table 11 also comments on assumed technology levels relative to today's attainable values.

No technology advancement beyond properties available today were assumed for the point design study or in the resulting algorithm.

### Summary of Anticipated Benefits with VCSC-RW TPTL Configuration

- Higher reliability
- Higher efficiency due to lower copper losses
- Increased power density due to improved utilization of PM material
- Smaller size, lower mass
- Optimum configuration to facilitate stator cooling using either direct flow [using liquid or gas] or by indirect outer jacket cooling flow.
- Competitive performance over a wide power range
- Higher operating speed allowables

#### 2.2.4.3 Algorithm Development

With the selection of the VCSC-RW TPTL configuration, fourteen valid point designs remained from which to formulate a conceptual design algorithm GENSIZE for turbo-generator systems. This data is contained in Tables 9 and 10 and represents seven power levels and two rotor surface speeds.

In order to develop the appropriate algorithms for size, mass and dimension, classical generator/motor scaling laws were applied to compute appropriate sizing coefficients. All algorithms considered design kVA, design voltage and rotor surface speed as the salient independent parameters. By applying the classical  $ND^2L$  [proportional to kVA] law the rotor diameter sizing coefficient could be determined. Overall dimensions [overall length and OD] were similarly converted to algorithm form. The four relevant equations contained in the generator sizing routine are as follows:

$$D_{\text{rotor}} = \left[ \left( \frac{U_{\text{tip}}}{700} \right)^{0.468} * (40.65 + 6.6E-4 * V * \left( \frac{U_{\text{tip}}}{700} \right)^{2.5}) * \text{kVA}^{0.075} \right] * \left[ \frac{\text{kVA}}{N * \{L/D\}_{\text{rotor}}} \right]^{1/3} \quad [1]$$

$$M_{\text{em}} = 1.938 * \left( \frac{U_{\text{tip}}}{700} \right)^{0.591} * (1.0467 - 3.3E-5 * V) * D_{\text{rotor}}^{2.85} * \left( \frac{\{L/D\}_{\text{rotor}} + .48}{2.98} \right) \quad [2]$$

$$D_{\text{stator}} = \left( \frac{U_{\text{tip}}}{700} \right)^{-0.4} * (2.14 - 0.12 * \text{kVA}^{0.175} - 2.25E-5 * V) * D_{\text{rotor}} \quad [3]$$

$$L_{\text{ola}} = (2.98 - .02 * D_{\text{rotor}}) * D_{\text{rotor}} * \left( \frac{\{L/D\}_{\text{rotor}} + .48}{2.98} \right) \quad [4]$$

Where;

$D_{\text{rotor}}$  = Rotor Outside Diameter [including sleeve], inches

$\{L/D\}_{\text{rotor}}$  = Rotor L/D; Magnet Length/Sleeve OD

$M_{\text{em}}$  = Generator Electro-magnetic Weight, lb<sub>m</sub>  
\* Copper and insulation  
\* Magnet and Sleeve  
\* Polyamide Structure  
\* Complete Flux Return Path Laminant

$D_{\text{stator}}$  = Generator Stator Outside Diameter, inches

$L_{\text{ola}}$  = Generator Overall Length, inches  
\* allowance for end turns/connections included

V = Generator Output Voltage, RMSv, line-to-line

kVA = Generator kiloVolt-Amperes as defined by Power and PF

$U_{\text{tip}}$  = Design Generator Surface Speed, ft/sec

#### Algorithm Validation

Tables 9 and 10 also contain mass and dimensional data computed from the equations above. The values computed from the developed algorithms generally agree within a few percent with the point design values and represent attainable designs which can be built with today's technology.

Details of routine function and assumptions are available from the code annotation contained Appendix B in the subroutine GENSIZE.

### 3.0 Conclusions and Recommendations

The closed Brayton cycle computer model described in this document was developed for inclusion into the NASA LeRC overall Nuclear Electric Propulsion (NEP) end-to-end systems model. The overall code will predict system and subsystem performance to significant detail. The code is intended to provide greater depth to the NEP system modelling which is required to more accurately predict the impact of specific technology on system performance.

The closed Brayton cycle power conversion system computer model is parametrically based to allow for conducting detailed optimization studies and to provide for easy integration into an overall optimizer driver routine. The power conversion model includes the modelling of the turbines, alternators, compressors, ducting, and heat exchangers (hot-side heat exchanger and recuperator). The system characteristics determined include estimates of mass, efficiency, and the characteristic dimensions of the major power conversion system components. These characteristics are parametrically modelled as a function of input parameters such as the aerodynamic configuration, turbine inlet temperature, cycle temperature ratio, power level, lifetime, materials, and redundancy. The model provides for selecting either an axial or radial aerodynamic configuration with the turbine inlet temperature limited to about 1500 K due to material considerations. The code is capable of evaluating the system performance over a temperature ratio ranging from 2.5 to 4.0. The model is designed for systems with a power level ranging from 50 to 2000 kWe using the Two-pole Toothless (TPTL) alternator as the baseline. The model also accounts for the desired system lifetime in the system characterization with an acceptable range from 2 to 10 years.

The closed Brayton cycle parametric model provides a useful tool for doing detailed system optimizations and parametric analysis. The code developed, although quite large, is relatively simple to use allowing for expedient parametric analysis. In the overall context of the NEP systems model, the Brayton parametric code will provide useful design information for developing an optimized Brayton system for specific applications and for assessing the impact of technology on the system performance.

**Appendix A**  
**Summary of Global Nomenclature in Brayton System Model**

## Summary of Global Nomenclature in Brayton System Model

AFINs	REAL	Recup. Fin Area/Volume; s=L or H for Loc., sq in/cu in
ALPHAs	REAL	Recup. Total Heat Tr. Area/Vol; s=L or H for Loc., dmls
ANSQC	REAL	Flow Area x Speed^2 for Axial Compressors, sq in-rpm^2
ANSQT	REAL	Flow Area x Speed^2 for Axial Turbines, sq in-rpm^2
APLATs	REAL	Recup. Plate Area/Vol.; s=L or H for Loc., sq in/cu in
BETA	REAL	Turbine Pressure Ratio/Compressor Pressure Ratio, dmls
BRGLOSS	REAL	Rotor Bearing Parasitic Loss, kWs
CLNTTYPE	CH*10	Generator Coolant Descriptor, e.g., 'N-HEPTANE'
COE	REAL	Current Rotor Sizing Coefficient [=f(TIPSPDG,KVA)]
COMFLO	REAL	Compressor Inlet Flowrate, 1bm/sec
COMPDIA	REAL	Radial Compressor Wheel Tip Diameter, inches
COMPSS	REAL	Compressor Specific Speed
COMPTYPE	CH*10	Compressor Type, either 'AXIAL' or 'RADIAL'
COOLING	REAL	Gen. Induced Cooling Load [EM + bearing + windage], kWt
CP	REAL	Working Fluid Specific Heat [Cp], BTU/(1bm-R)
CPCLNT	REAL	Coolant Specific Heat, BTU/(1bm-R)
CYCDES(8)	CH*10	Descriptive Title for Current Case
DGENRTR	REAL	Generator Rotor Diameter Including Sleeve, inches
DGENSTR	REAL	Generator Stator Diameter Including Sleeve, inches
DPMH	REAL	DP [pressure loss] Design Margin for Recuperator High Pressure Side, dmls
DPML	REAL	DP [pressure loss] Design Margin for Recuperator Low Pressure Side, dmls
DPOPi	REAL	Fractional Pressure Loss for Section Ending at Statepoint "i", kWt
DPOPREC	REAL	Total Recuperator [both passages] Fractional Pressure Loss, dmls
EFFC	REAL	Compressor Adiabatic Efficiency, dmls
EFFCYCLE	REAL	Cycle Efficiency=GROSSEP/Net Thermal Input to Gas, dmls
EFFT	REAL	Turbine Adiabatic Efficiency, dmls
EFFR	REAL	Recuperator , dmls
ERRORC	CH*64	Compressor Subroutine Diagnostic Error Message
ERRORF	CH*64	Fluid Properties Subroutine Diagnostic Error Message
ERRORG	CH*64	Generator Subroutine Diagnostic Error Message
ERRORM	CH*64	Recuperator Subroutine Diagnostic Error Message
ERRORT	CH*64	Turbine Subroutine Diagnostic Error Message
ETACOMP	REAL	Compressor Adiabatic Efficiency, dmls
ETATURB	REAL	Turbine Adiabatic Efficiency, dmls
FLOW(i)	REAL	Working Fluid Flowrate at Statepoint "i", 1bm/sec
GAMMA	REAL	Gas Ratio of Specific Heats [Cv/Cp], dmls
GAS(2)	CH*10	Working Fluid Constituent Gases, e.g., 'XENON', 'HELIUM'
GENASP	REAL	Generator Rotor Aspect Ratio, Sleeve OD/Magnet Length
GENTYPE	CH*20	Generator Type Descriptor 'RING WOUND TPTL PMG'
GROSSEP	REAL	Gross Electric Power at Generator Terminals, kWt
HDs	REAL	Recup. Hydraulic Diam.; s=L or H for Loc., in.
INTTYPE	CH*20	Electrical Interface Descriptor, either 'TRANSFORMER' or 'RECTIFIER'
IPRINT	INTEG	Output File [PROG\$OUT] Switch, 0=OFF, 1,2,3= Increasing Detail in Output File

## Summary of Global Nomenclature in Brayton System Model (continued)

ISTGC	INTEG	Number of Axial Compressor Stages, dm <sub>ls</sub>
ISTGT	INTEG	Number of Axial Turbine Stages, dm <sub>ls</sub>
KFINS	REAL	Recup. Fin Thermal Conductivity; s=L or H for Loc., BTU/(hr-ft-R)[input] & BTU/(sec-ft-R)[code]
KVA	REAL	Generator Output, kVA
LEQs	REAL	Recuperator Fin Equiv Length; s=L or H for Loc., in.
LGENTOT	REAL	Overall Length of Generator EM Section, inches
LIFETIME	REAL	Generator Design Lifetime, years
MASSEM	REAL	Estimated Generator Electro-magnetic Mass, lbm
MASSGEN	REAL	Estimated Generator Mass w/ Integration Allowance, lbm
MOLWT	REAL	Working Fluid Molecular Weight, lbm/(lb-mol)
NETSP	REAL	Net Shaft Power [turb-comp-bearings-windage], kWs
NFINS	REAL	Recuperator Fins/inch; s= L or H for Location
OLDTS	REAL	Old Tip Speed Value [used for convergence test], ft/sec
PI	REAL	3.14159265
PITCH	REAL	Repeating Sandwich Thickness in Recuperator Core, in.
POWER	REAL	Generator Terminal Power, kW <sub>e</sub>
PR(i)	REAL	Working Fluid Pressure Ratio Between Statepoints "i-1" and "i"
PRC	REAL	Compressor Pressure Ratio [total-to-total], dm <sub>ls</sub>
PRESS(i)	REAL	Working Fluid Pressure at Statepoint "i", psia
PREXP	REAL	Prandtl No. Exponent Used in Heat Transfer Calcs., dm <sub>ls</sub>
PRNDTL	REAL	Current Heat Transfer "Prandtl No.", dm <sub>ls</sub>
PWFCTR	REAL	Load Power Factor at Generator Terminals [lagging]
QHSA	REAL	Heat Source Gross Thermal Power, kW <sub>t</sub>
QLOSSi	REAL	Thermal Loss for Section Ending at Statepoint "i", kW <sub>t</sub>
RADHC	REAL	Axial Compressor Flowpath Hub Radius, inches
RAD1C	REAL	Axial Compressor Flowpath Inlet Tip Radius, inches
RADNC	REAL	Axial Compressor Flowpath Discharge Tip Radius, inches
RADHT	REAL	Axial Turbine Flowpath Hub Radius, inches
RAD1T	REAL	Axial Turbine Flowpath Inlet Tip Radius, inches
RADNT	REAL	Axial Turbine Flowpath Discharge Tip Radius, inches
RECH	REAL	Recuperator Core Height, inches
RECLC	REAL	Recuperator Core Length, inches
RECLOA	REAL	Recuperator Overall Length, inches
RECW	REAL	Recuperator Core Width, inches
RHOFINS	REAL	Recup. Fins Density; s=L or H for Location, lbm/cu in
ROPLAT	REAL	Recuperator Splitter Plate Density, lbm/cu in
ROBRAZ	REAL	Recuperator Braze Density, lbm/cu in
SIGMAS	REAL	Recup. Flow:Frontal Area Ratio; s=L or H for Loc., dm <sub>ls</sub>
SPACEs	REAL	Recuperator Plate Spacing; s=L or H for Location, in.
SPEED	REAL	Generator Design Rotating Speed, rpm
TBRAZE	REAL	Recuperator Braze Thickness [4/pitch], inches
TEMP(i)	REAL	Working Fluid Temperature at Statepoint "i", deg R
TFINS	REAL	Recuperator Fins Thickness; s=L or H for Location, in.
TINCLNT	REAL	Specified Generator Coolant Inlet Temperature, deg R
TIPSPDG	REAL	Generator Rotor Surface Speed, ft/sec [limit is 700]
TITLER(8)	CH*10	Description of Recuperator Core Matrix Geometry
TOUTCLNT	REAL	Specified Generator Coolant Outlet Temperature, deg R
TPLATE	REAL	Recuperator Splitter Plate Thickness, inches

## Summary of Global Nomenclature in Brayton System Model (continued)

TURBDIA	REAL	Radial Turbine Wheel Tip Diameter, inches
TURBSS	REAL	Turbine Specific Speed
TURBTYPE	CH*10	Turbine Type, either 'AXIAL' or 'RADIAL'
TVAR(18)	CH*10	Variable Names [as converted in SUBROUTINE SETUP]
UAMH	REAL	UA [heat transfer] DesignMargin for Recuperator High Pressure Side, dmls
UAML	REAL	UA [heat transfer] DesignMargin for Recuperator Low Pressure Side, dmls
UTIPC	REAL	Compressor [maximum for axials] Tip Speed, ft/sec
UTIPT	REAL	Turbine [maximum for axials] Tip Speed, ft/sec
VOLTAGE	REAL	Desired Generator Output Voltage, 3Ph, line-line, RMS
WARNINGC	CH*64	Compressor Subroutine Diagnostic Warning Message
WARNINGF	CH*64	Fluid Properties Subroutine Diagnostic Warning Message
WARNINGG	CH*64	Generator Subroutine Diagnostic Warning Message
WARNINGR	CH*64	Recuperator Subroutine Diagnostic Warning Message
WARNINGT	CH*64	Turbine Subroutine Diagnostic Warning Message
WCLNT	REAL	Calculated Coolant Flow Rate, 1bm/sec
WGTDUCT(6)	REAL	Mass of the Individual Duct Segments, 1bm
WDCTTOT	REAL	Total of Duct Masses, 1bm
WGTHRA	REAL	Heat Rejection Assembly Mass, 1bm
WGTHSA	REAL	Heat Source Assembly Mass, 1bm
WTREC	REAL	Recuperator Assembly Mass, 1bm
WGTTAC	REAL	Turbo-alternator Assembly Mass, 1bm
WGTTOT	REAL	Total PCU Mass, 1bm
WINDAGE	REAL	Generator Main Gap Windage Loss, kWs
XBLC	REAL	Bleed Fraction of Flow(1) Returned to Statepoint #2 [comp wheel tip], fraction
XBLT	REAL	Bleed Fraction of Flow(1) Returned to Statepoint #13 [turbine stage exit], fraction

**Appendix B**

**Source Code Listing for**  
**Closed-Brayton Cycle Parametric Design Program**

**BRAY1.FOR**

```

$DEBUG
$NOTRUNCATE
      PROGRAM MANE
C **** MANE is checked-out and final on October 6, 1992

      IMPLICIT REAL (A-H)
      IMPLICIT INTEGER (I-J)
      IMPLICIT REAL (K-Z)
      REAL JCON
      CHARACTER*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18),
& COMPTYPE,TURBTYPE,CLNTTYPE
      CHARACTER*20 GENTYPE,INTTYPE
      CHARACTER*64 ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,WARNIGC,
& WARNIGR,WARNIGG

      COMMON/DIAGNOS/ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,
& WARNIGC,WARNIGR,WARNIGG
      COMMON/OUTP/WGTTOT,WGTTAC,RECLC,RECLOA,RECH,RECW

      COMMON /DUCTI1/ IDUCT, PIN, POUT, MDOT, TIN, MW
      COMMON/DUCTI2/ EL1(6), EL2(6), TWL(6), DNWL(6), NMFI(6),
& TMFI(6), DMFI(6), ETA(6)
      COMMON /DUCTOUT/ WGTDUCT(6), DIAM(6), LGTH(6), VGAS(6)
      COMMON/XDUCT/ XTWL(6), XDNWL(6), XTMFI(6), XDMFI(6)
      COMMON/HSACALCS/ QHSA,WGTHSA,DP9

      COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
      COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,
& CPCLNT,LIFETIME
      COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
& RADNC,ANSQC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,
& RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT
      COMMON/OPTIM/ VAR(18),TVAR,IPRINT
      COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
      COMMON/TITLE/ CYCDES,TITLER

      COMMON/SI/XVAR(17), XTEMP(17)

      COMMON/LOSS/ QLOSS8,QLOSS9,QLOSS10,QLOSS13,FCTQ8,FCTQ10,FCTQ13
      COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFL0,
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,
& DPOP16,DPOP17,NETSP,GROSSE,PEFFCYCLE,WINDAGE
      COMMON/GASBRG/BRGLOSS
      COMMON/RECUP/ DPOPREC,UAML,UAMH,DPML,DPMH,
& PREXP,NFINL,TFINL,SPACEL,LEQL,ROFINL,KFINL,NFINH,TFINH,SPACEH,
& LEQH,ROFINH,KFINH,TPLATE,ROPLAT,TBRAZE,ROBRAZ,
& SIGMAL,ALPHAL,AFINL,APlATL,HDL,SIGMAH,ALPHAH,AFINH,APlATH,HDH,
& PITCH,WGTREC
      COMMON/MASSES/MASSCMP,MASSTRB,MWHLC,MWHLT,MSFTC,MSFTT,MCASEC,
& MCASET,MDIAC,MDIAT

```

COMMON/IHX/ DPSHELL1,ANTUBES1,DPTUBE1,DTOL21,ALSHEL1,AMSHELL1,  
 & AMPLATES1,AMTUBES1,AMINSUL1,AMHEADS1,AMSTRT1,ANETMASS1,XMNHEX1,  
 & HSHELL1,AFRIC1,UNEW1,RETUBE1,THC1,AMTSHT1,PRATIO, IHXFLG, UEST1,  
 & THIN1, THOUT1, TTUBE1, ANPLATES1,AKTUBE1, TINS1, DENINS1,  
 & DENSSH1, DTUBE1

COMMON/CONST/PI,RU,GO,JCON,RZERO

C\*\*\*\* assign constants for later computational use  
 PI=3.14159265  
 RU=1545.  
 GO=32.1739  
 JCON=778.16  
 RZERO=0.

C \*\*\*\*\* Nomenclature \*\*\*\*\*

C	AFINs	REAL	Recup. Fin Area/Volume; s=L or H for Loc., sq in/cu in
C	ALPHAs	REAL	Recup. Total Heat Tr. Area/Vol; s=L or H for Loc., dmls
C	ANSQC	REAL	Flow Area x Speed^2 for Axial Compressors, sq in-rpm^2
C	ANSQCL	REAL	Approximate AN^2 Limit for Axial Compressors, sq in-rpm^2
C	ANSQT	REAL	Flow Area x Speed^2 for Axial Turbines, sq in-rpm^2
C	ANSQTL	REAL	Approximate AN^2 Limit for Axial Turbines, sq in-rpm^2
C	APLATs	REAL	Recup. Plate Area/Vol.; s=L or H for Loc., sq in/cu in
C	BETA	REAL	Turbine Pressure Ratio/Compressor Pressure Ratio, dmls
C	BRGLOSS	REAL	Rotor Bearing Parasitic Loss, kWs
C	CLNTTYPE	CH*10	Generator Coolant Descripptor, e.g., 'N-HEPTANE'
C	COE	REAL	Current Rotor Sizing Coefficient [=f(TIPSPDG,KVA)]
C	COMFLO	REAL	Compressor Inlet Flowrate, 1bm/sec
C	COMPDIA	REAL	Radial Compressor Wheel Tip Diameter, inches
C	COMPSS	REAL	Compressor Specific Speed
C	COMPTYPE	CH*10	Compressor Type, either 'AXIAL' or 'RADIAL'
C	COOLING	REAL	Generator Induced Cooling Load [EM+windage+beargs], kWt
C	CP	REAL	Working Fluid Specific Heat [Cp], BTU/(1bm-R)
C	CPLCNT	REAL	Coolant Specific Heat, BTU/(1bm-R)
C	CYCDES(8)	CH*10	Descriptive Title for Current Case
C	DGENRTR	REAL	Generator Rotor Diameter Including Sleeve, inches
C	DGENSTR	REAL	Generator Stator Diameter Including Sleeve, inches
C	DPMH	REAL	DP [pressure loss] Design Margin for Recuperator High Pressure Side, dmls
C	DPML	REAL	DP [pressure loss] Design Margin for Recuperator Low Pressure Side, dmls
C	DPOPi	REAL	Fractional Pressure Loss for Section Ending at Statepoint "i", kWt
C	DPOPREC	REAL	Total Recuperator [both passages] Fractional Pressure Loss, dmls
C	EFFC	REAL	Compressor Adiabatic Efficiency, dmls
C	EFFCYCLE	REAL	Cycle Efficiency=GROSSE/Net Thermal Input to Gas, dmls
C	EFFT	REAL	Turbine Adiabatic Efficiency, dmls
C	EFFR	REAL	Recuperator , dmls
C	ERRORC	CH*64	Compressor Subroutine Diagnostic Error Message
C	ERRORF	CH*64	Fluid Properties Subroutine Diagnostic Error Message

C	ERRORG	CH*64	Generator Subroutine Diagnostic Error Message
C	ERRORR	CH*64	Recuperator Subroutine Diagnostic Error Message
C	ERRORT	CH*64	Turbine Subroutine Diagnostic Error Message
C	ETACOMP	REAL	Compressor Adiabatic Efficiency, dmls
C	ETATURB	REAL	Turbine Adiabatic Efficiency, dmls
C	FLOW(i)	REAL	Working Fluid Flowrate at Statepoint "i", 1bm/sec
C	GAMMA	REAL	Gas Ratio of Specific Heats [Cv/Cp], dmls
C	GAS(2)	CH*10	Working Fluid Constituent Gases, e.g., 'XENON', 'HELIUM'
C	GENASP	REAL	Generator Rotor Aspect Ratio, Sleeve OD/Magnet Length
C	GENTYPE	CH*20	Generator Type Descriptor 'RING WOUND TPTL PMG'
C	GROSSEP	REAL	Gross Electric Power at Generator Terminals, kWe
C	HDs	REAL	Recup. Hydraulic Diam.; s=L or H for Loc., in.
C	INTTYPE	CH*20	Electrical Interface Descriptor, either 'TRANSFORMER' or 'RECTIFIER'
C	IPRINT	INTEG	Output File [PROG\$OUT] Switch, 0=OFF, 1,2,3= Increasing Detail in Output File
C	ISTGC	INTEG	Number of Axial Compressor Stages, dmls
C	ISTGT	INTEG	Number of Axial Turbine Stages, dmls
C	KFINS	REAL	Recup. Fin Thermal Conductivity; s=L or H for Loc., BTU/(hr-ft-R)[input] & BTU/(sec-ft-R)[code]
C	KVA	REAL	Generator Output, kVA
C	LEQs	REAL	Recuperator Fin Equiv Length; s=L or H for Loc., in.
C	LGENTOT	REAL	Overall Length of Generator EM Section, inches
C	LIFETIME	REAL	Generator Design Lifetime, years
C	MASSEM	REAL	Estimated Generator Electro-magnetic Mass, 1bm
C	MASSEGEN	REAL	Estimated Generator Mass w/ Integration Allowance, 1bm
C	MOLWT	REAL	Working Fluid Molecular Weight, 1bm/(lb-mol)
C	NETSP	REAL	Net Shaft Power [turb-comp-bearings-widge], kW
C	NFINS	REAL	Recuperator Fins/inch; s= L or H for Location
C	OLDTD	REAL	Old Tip Speed Value [used for convergence test], ft/sec
C	PI	REAL	3.14159265
C	PITCH	REAL	Repeating Sandwich Thickness in Recuperator Core, in.
C	POWER	REAL	Generator Terminal Power, kWe
C	PR(i)	REAL	Working Fluid Pressure Ratio Between Statepoints "i-1" and "i"
C	PRC	REAL	Compressor Pressure Ratio [total-to-total], dmls
C	PRESS(i)	REAL	Working Fluid Pressure at Statepoint "i", psia
C	PREXP	REAL	Prandtl No. Exponent Used in Heat Transfer Calcs., dmls
C	PRNDTL	REAL	Current Heat Transfer "Prandtl No.", dmls
C	PWRFCTR	REAL	Load Power Factor at Generator Terminals [lagging]
C	QHSA	REAL	Heat Source Gross Thermal Power, kW
C	QLOSSi	REAL	Thermal Loss for Section Ending at Statepoint "i", kwt
C	RADHC	REAL	Axial Compressor Flowpath Hub Radius, inches
C	RAD1C	REAL	Axial Compressor Flowpath Inlet Tip Radius, inches
C	RADNC	REAL	Axial Compressor Flowpath Discharge Tip Radius, inches
C	RADHT	REAL	Axial Turbine Flowpath Hub Radius, inches
C	RADIT	REAL	Axial Turbine Flowpath Inlet Tip Radius, inches
C	RADNT	REAL	Axial Turbine Flowpath Discharge Tip Radius, inches
C	RECH	REAL	Recuperator Core Height, inches
C	RECLC	REAL	Recuperator Core Length, inches
C	RECLOA	REAL	Recuperator Overall Length, inches

C	RECW	REAL	Recuperator Core Width, inches
C	RHOFINS	REAL	Recup. Fins Density; s=L or H for Location, 1bm/cu in
C	ROPLAT	REAL	Recuperator Splitter Plate Density, 1bm/cu in
C	ROBRAZ	REAL	Recuperator Braze Density, 1bm/cu in
C	SIGMAs	REAL	Recup. Flow:Frontal Area Ratio; s=L or H for Loc., dmls
C	SPACEs	REAL	Recuperator Plate Spacing; s=L or H for Location, in.
C	SPEED	REAL	Generator Design Rotating Speed, rpm
C	TBRAZE	REAL	Recuperator Braze Thickness [4/pitch], inches
C	TEMP(i)	REAL	Working Fluid Temperature at Statepoint "i", deg R
C	TFINs	REAL	Recuperator Fins Thickness; s=L or H for Location, in.
C	TINCLNT	REAL	Specified Generator Coolant Inlet Temperature, deg R
C	TIPSPDG	REAL	Generator Rotor Surface Speed, ft/sec [limit is 700]
C	TITLER(8)	CH*10	Description of Recuperator Core Matrix Geometry
C	TOUTCLNT	REAL	Specified Generator Coolant Outlet Temperature, deg R
C	TPLATE	REAL	Recuperator Splitter Plate Thickness, inches
C	TURBDIA	REAL	Radial Turbine Wheel Tip Diameter, inches
C	TURBSS	REAL	Turbine Specific Speed
C	TURBTYPE	CH*10	Turbine Type, either 'AXIAL' or 'RADIAL'
C	TVAR(18)	CH*10	Variable Names [as converted in SUBROUTINE SETUP]
C	UAMH	REAL	UA [heat transfer] DesignMargin for Recuperator High Pressure Side, dmls
C	UAML	REAL	UA [heat transfer] DesignMargin for Recuperator Low Pressure Side, dmls
C	UTIPC	REAL	Compressor [maximum for axials] Tip Speed, ft/sec
C	UTIPT	REAL	Turbine [maximum for axials] Tip Speed, ft/sec
C	VOLTAGE	REAL	Desired Generator Output Voltage, 3Ph, Line-line, RMS
C	WARNINGC	CH*64	Compressor Subroutine Diagnostic Warning Message
C	WARNINGF	CH*64	Fluid Properties Subroutine Diagnostic Warning Message
C	WARNINGG	CH*64	Generator Subroutine Diagnostic Warning Message
C	WARNINGR	CH*64	Recuperator Subroutine Diagnostic Warning Message
C	WARNINGT	CH*64	Turbine Subroutine Diagnostic Warning Message
C	WCLNT	REAL	Calculated Coolant Flow Rate, 1bm/sec
C	WGTDUCT(6)	REAL	Mass of the Individual Duct Segments, 1bm
C	WDCTTOT	REAL	Total of Duct Masses, 1bm
C	WGTHRA	REAL	Heat Rejection Assembly Mass, 1bm
C	WGTHSA	REAL	Heat Source Assembly Mass, 1bm
C	WGTREC	REAL	Recuperator Assembly Mass, 1bm
C	WGTTAC	REAL	Turbo-alternator Assembly Mass, 1bm
C	WGTTOT	REAL	Total PCU Mass, 1bm
C	WINDAGE	REAL	Generator Main Gap Windage Loss, kWs
C	XBLA	REAL	Bleed Fraction of Flow(1) Returned to Statepoint #2 [comp wheel tip], fraction
C	XBLT	REAL	Bleed Fraction of Flow(1) Returned to Statepoint #13 [turbine stage exit], fraction

C \*\*\*\*\* Statepoint Definition \*\*\*\*\*

C Statepoint                  Location

C Note: Total [stagnation] conditions except as noted.

C # 1 Compressor Inlet

```

C
C Note: Statepoints # 2, 3, 4 are relevant with radial compressors only
C
C   # 2  Compressor Rotor Tip [static pressure]
C   # 3  Compressor Diffuser Inlet
C   # 4  Compressor Diffuser Exit [comp-end bleed flow added]
C   # 5  Compressor Stage Exit [comp-end bleed flow removed]
C   # 6  Recuperator High Pressure Flow Inlet
C   # 7  Recuperator High Pressure Flow Discharge
C   # 8  HSA Inlet
C   # 9  HSA Discharge
C   # 10 Turbine Inlet
C
C Note: Statepoints # 11 is relevant with radial turbines only
C
C   # 11 Turbine Nozzel Exit
C   # 12 Turbine Discharge
C   # 13 Recuperator Low Pressure Flow Inlet [turb-end bleed flow added]
C   # 14 Recuperator Low Pressure Flow Discharge
C   # 15 HRA Inlet
C   # 16 HRA Discharge
C   # 17 Compressor Inlet Duct Exit [= to Statepoint #1]
C

```

```

OPEN (60,FILE='CYC$INP',FORM='FORMATTED',STATUS='OLD',
&ACCESS='SEQUENTIAL')

```

```
C **** Read from data file the Brayton Cycle operating parameters
```

```

READ (60,10) (CYCDES(I),I=1,8)
READ (60,20) IPRINT,POWER,PWRFCTR,VOLTAGE,GENASP,XTEMP(9)
READ (60,30) COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
READ (60,40) GAS(1),GAS(2),CPCLNT,XTINCLNT,XTOUTCLNT,XBLC,XBLT
READ (60,50) LIFETIME,FCTQ8,FCTQ10,FCTQ13
READ (60,10) (TVAR(I),I=1,15)
READ (60,50) (VAR(I),I=1,15)
10 FORMAT (8A10)
20 FORMAT (I5,5X,5E10.3)
30 FORMAT (2A10,2A20,A10)
40 FORMAT (2A10,5E10.3)
50 FORMAT (8E10.3)
CLOSE (60)

```

```
C *** Convert to english units for use in the program ***
```

```

TEMP(9)=XTEMP(9)*1.8
TINCLNT=XTINCLNT*1.8
TOUTCLNT=XTOUTCLNT*1.8
XVAR(1)=VAR(1)
VAR(1)=XVAR(1)*1.8

```

```
IF (IPRINT.GT.0) OPEN (61,FILE='PROG$OUT',FORM='FORMATTED',
```

```

& STATUS='OLD', ACCESS='SEQUENTIAL')

C **** Read from data file the recuperator geometry definition.

OPEN (60,FILE='REC$INP',FORM='FORMATTED',STATUS='OLD',
&ACCESS='SEQUENTIAL')
CALL RECINP
CLOSE (60)

C **** Read data file for the IHX ****
OPEN (60,FILE='IHX$INP',FORM='FORMATTED',STATUS='OLD',
&ACCESS='SEQUENTIAL')

C Read input data from data file "IHX$INP"

READ (60,95) IHXFLG, XUEST1, XTHIN1, XTHOUT1, XTTUBE1, ANPLATES1
READ (60,96) XAKTUBE1, XTINS1, XDENINS1, XDENSSH1, XDTUBE1

95 FORMAT (I10,5F10.5)
96 FORMAT (5F10.5)

C ** CONVERT TO ENGINEERING UNITS FOR CALCULATIONS

UEST1=XUEST1*3600.*(0.3048**2.)/(1054.4*1.8)
THIN1=XTHIN1*1.8
THOUT1=XTHOUT1*1.8
TTUBE1=XTTUBE1/2.54
AKTUBE1=XAKTUBE1*3600.*0.3048/(1054.4*1.8)
TINS1=XTINS1/2.54
DENINS1=XDENINS1*(2.54**3.)*(12.**3.)/454.
DENSSH1=XDENSSH1*(2.54**3.)*(12.**3.)/454.
DTUBE1=XDTUBE1/2.54

CLOSE(60)

C ****
C If required, read data file for heat rejection gas cooler

C OPEN (60,FILE='GC$INP',FORM='FORMATTED',STATUS='OLD',
C &ACCESS='SEQUENTIAL')
C CLOSE(60)

C ****

OPEN (60,FILE='DUCT$INP',FORM='FORMATTED',STATUS='OLD',
&ACCESS='SEQUENTIAL')

C Read input data from data file "DUCT$INP"

READ (60,100) (EL1(I), I=1,6)

```

```

READ (60,100) (EL2(I), I=1,6)
READ (60,200) (XTWL(I), I=1,6)
READ (60,200) (XDNWL(I), I=1,6)
READ (60,100) (NMFI(I), I=1,6)
READ (60,200) (XTMFI(I), I=1,6)
READ (60,200) (XDMFI(I), I=1,6)
READ (60,200) (ETA(I), I=1,6)

C *** Convert to engineering units for calculations
DO 98 I=1,6
  TWL(I)=XTWL(I)/2.54
  DNWL(I)=XDNWL(I)*(1./453.6)*(2.54**3)
  TMFI(I)=XTMFI(I)/2.54
  DMFI(I)=XDMFI(I)*(1./453.6)*(2.54**3)
98 CONTINUE

100 FORMAT (6F10.3)
200 FORMAT (6F10.5)

CLOSE(60)

```

C \*\*\*\* Call the Brayton system and component design routines.

CALL BRAYTON

END

#### SUBROUTINE BRAYTON

```

C **** BRAYTON is checked-out and final on October 6, 1992
$DEBUG
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
REAL JCON

CHARACTER*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18),
& COMPTYPE,TURBTYPE,CLNTTYPE
CHARACTER*20 GENTYPE,INTTYPE
CHARACTER*64 ERRORT,ERRORC,ERRORR,ERRORG,ERRORF,WARNIGT,WARNIGC,
& WARNIGR,WARNIGG
CHARACTER*60 ERR1, ERR2, ERR3, ERR4, ERR5, ERR6, ERR7, ERR8,
& ERR9, ERR10, ERR11, ERR12

COMMON/DIAGNOS/ERRORT,ERRORC,ERRORR,ERRORG,ERRORF,WARNIGT,
& WARNIGC,WARNIGR,WARNIGG
COMMON/OUTP/WGTTOT,WGTTAC,RECLC,RECLOA,RECH,RECW
COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,
& CPCLNT,LIFETIME
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
```

& RADNC, ANSQC, ANSQCL, COMPSS, ISTGC, ETATURB, TURBDIA, UTIPT,  
 & RADHT, RADIT, RADNT, ANSQT, ANSQL, TURBSS, ISTGT  
 COMMON/OPTIM/ VAR(18), TVAR, IPRINT  
 COMMON/FLUID/ GAS, GAMMA, MOLWT, CP, PRNDTL  
 COMMON/TITLE/ CYCDES, TITLER  
  
 COMMON /DUCTI1/ IDUCT, PIN, POUT, MDOT, TIN, MW  
 COMMON/DUCTI2/ EL1(6), EL2(6), TWL(6), DNWL(6), NMFI(6),  
 & TMFI(6), DMFI(6), ETA(6)  
 COMMON /DUCTOUT/ WGTDUCT(6), DIAM(6), LGTH(6), VGAS(6)  
 COMMON /DUCTERR/ ERR1, ERR2, ERR3, ERR4, ERR5, ERR6, ERR7, ERR8,  
 & ERR9, ERR10, ERR11, ERR12  
  
 COMMON/HSACALCS/ QHSA, WGTHSA, DP9  
 COMMON/HRACALCS/ QHRA, WGTHRA, DP16  
 COMMON/LOSS/ QLOSS8, QLOSS9, QLOSS10, QLOSS13, FCTQ8, FCTQ10, FCTQ13  
 COMMON/CYCLE/TEMP(17), PRESS(17), FLOW(17), PR(17), BETA, COMFLO,  
 & PRC, EFFC, EFFT, EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,  
 & DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,  
 & DPOP16,DPOP17,NETSP,GROSSEP,EFFCYCLE,WINDAGE  
 COMMON/GASBRG/BRGLOSS  
 COMMON/RECUP/ DPOPREC, UAML, UAMH, DPML, DPMH,  
 & PREXP, NFINL, TFINL, SPACEL, LEQL, ROFINL, KFINL, NFINH, TFINH, SPACEH,  
 & LEQH, ROFINH, KFINH, TPLATE, ROPLAT, TBRAZE, ROBRAZ,  
 & SIGMAL, ALPHAL, AFINL, APLATL, HDL, SIGMAH, ALPHAH, AFINH, APLATH, HDH,  
 & PITCH, WGTREC  
 COMMON/MASSES/MASSCMP, MASSTRB, MWHLC, MWHLT, MSFTC, MSFTT, MCASEC,  
 & MCASET, MDIAC, MDIAT  
 COMMON/IHX/ DPSHELL1, ANTUBES1, DPTUBE1, DOTL21, ALSHEL1, AMSHELL1,  
 & AMPLATES1, AMTUBES1, AMINSUL1, AMHEADS1, AMSTRT1, ANETMASS1, XMNHEX1,  
 & HSHELL1, AFRIC1, UNEW1, RETUBE1, THC1, AMTSHT1, PRATIO, IHXFLG, UEST1,  
 & THIN1, THOUT1, TTUBE1, ANPLATES1, AKTUBE1, TINS1, DENINS1,  
 & DENSSH1, DTUBE1  
  
 COMMON/CONST/PI, RU, GO, JCON, RZERO

C \*\*\*\*\* Nomenclature \*\*\*\*\*

C	AFINs	REAL	Recup. Fin Area/Volume; s=L or H for Loc., sq in/cu in
C	ALPHAs	REAL	Recup. Total Heat Tr. Area/Vol; s=L or H for Loc., dmls
C	ANSQCL	REAL	Approximate AN^2 Limit for Axial Compressors, sq in-rpm^2
C	ANSQT	REAL	Flow Area x Speed^2 for Axial Turbines, sq in-rpm^2
C	ANSQL	REAL	Approximate AN^2 Limit for Axial Turbines, sq in-rpm^2
C	APLATS	REAL	Recup. Plate Area/Vol.; s=L or H for Loc., sq in/cu in
C	BETA	REAL	Turbine Pressure Ratio/Compressor Pressure Ratio, dmls
C	BRGLOSS	REAL	Rotor Bearing Parasitic Loss, kWs
C	CLNTTYPE	CH*10	Generator Coolant Descrippor, e.g., 'N-HEPTANE'
C	COE	REAL	Current Rotor Sizing Coefficient [=f(TIPSPDG,KVA)]
C	COMFLO	REAL	Compressor Inlet Flowrate, lbm/sec
C	COMPDIA	REAL	Radial Compressor Wheel Tip Diameter, inches
C	COMPSS	REAL	Compressor Specific Speed
C	COMPTYPE	CH*10	Compressor Type, either 'AXIAL' or 'RADIAL'

C	COOLING	REAL	Generator Induced Cooling Load [EM + windage], kWt
C	CP	REAL	Working Fluid Specific Heat [Cp], BTU/(1bm-R)
C	CPCLNT	REAL	Coolant Specific Heat, BTU/(1bm-R)
C	CYCDES(8)	CH*10	Descriptive Title for Current Case
C	DIAM(6)	REAL	Inside diameter of individual duct segments
C	DGENRTR	REAL	Generator Rotor Diameter Including Sleeve, inches
C	DGENSTR	REAL	Generator Stator Diameter Including Sleeve, inches
C	DPMH	REAL	DP [pressure loss] Design Margin for Recuperator High Pressure Side, dmls
C	DPML	REAL	DP [pressure loss] Design Margin for Recuperator Low Pressure Side, dmls
C	DPOPi	REAL	Fractional Pressure Loss for Section Ending at Statepoint "i", kWt
C	DPOPREC	REAL	Total Recuperator [both passages] Fractional Pressure Loss, dmls
C	EFFC	REAL	Compressor Adiabatic Efficiency, dmls
C	EFFCYCLE	REAL	Cycle Efficiency=GROSSEP/Net Thermal Input to Gas, dmls
C	EFFT	REAL	Turbine Adiabatic Efficiency, dmls
C	EFFR	REAL	Recuperator , dmls
C	ERRORC	CH*64	Compressor Subroutine Diagnostic Error Message
C	ERRORF	CH*64	Fluid Properties Subroutine Diagnostic Error Message
C	ERRORG	CH*64	Generator Subroutine Diagnostic Error Message
C	ERRORM	CH*64	Recuperator Subroutine Diagnostic Error Message
C	ERRORT	CH*64	Turbine Subroutine Diagnostic Error Message
C	ETACOMP	REAL	Compressor Adiabatic Efficiency, dmls
C	ETATURB	REAL	Turbine Adiabatic Efficiency, dmls
C	FCTQi	REAL	Thermal Loss at Statepoint "i", Fraction of Input Q
C	FLOW(i)	REAL	Working Fluid Flowrate at Statepoint "i", 1bm/sec
C	GAMMA	REAL	Gas Ratio of Specific Heats [Cv/Cp], dmls
C	GAS(2)	CH*10	Working Fluid Constituant Gases, e.g.,'XENON      ', 'HELIUM'
C	GENASP	REAL	Generator Rotor Aspect Ratio, Sleeve OD/Magnet Length
C	GENTYPE	CH*20	Generator Type Descriptor 'RING WOUND TPTL PMG'
C	GROSSEP	REAL	Gross Electric Power at Generator Terminals, kWe
C	HDs	REAL	Recup. Hydraulic Diam.; s=L or H for Loc., in.
C	INTTYPE	CH*20	Electrical Interface Descriptor, either 'TRANSFORMER' or 'RECTIFIER'
C	IPRINT	INTEG	Output File [PROG\$OUT] Switch, 0=OFF, 1=ON
C			Increasing Detail in Output File
C	ISTGC	INTEG	Number of Axial Compressor Stages, dmls
C	ISTGT	INTEG	Number of Axial Turbine Stages, dmls
C	KFINS	REAL	Recup. Fin Thermal Conductivity; s=L or H for Loc., BTU/(hr-ft-R)[input] & BTU/(sec-ft-R)[code]
C	KVA	REAL	Generator Output, kVA
C	LEQs	REAL	Recuperator Fin Equiv Length; s=L or H for Loc., in.
C	LGENTOT	REAL	Overall Length of Generator EM Section, inches
C	LGTH(6)	REAL	Length of individual duct segments, in
C	LIFETIME	REAL	Generator Design Lifetime, years
C	MASSEM	REAL	Estimated Generator Electro-magnetic Mass, 1bm
C	MASSGEN	REAL	Estimated Generator Mass w/ Integration Allowance, 1bm

C	MOLWT	REAL	Working Fluid Molecular Weight, 1bm/(1b-mol)
C	NETSP	REAL	Net Shaft Power [turb-comp-bearings-widage], kWs
C	NFINS	REAL	Recuperator Fins/inch; s= L or H for Location
C	OLDTS	REAL	Old Tip Speed Value [used for convergence test], ft/sec
C	PI	REAL	3.14159265
C	PITCH	REAL	Repeating Sandwich Thickness in Recuperator Core, in.
C	POWER	REAL	Generator Terminal Power, kW
C	PR(i)	REAL	Working Fluid Pressure Ratio Between Statepoints "i-1" and "i"
C	PRC	REAL	Compressor Pressure Ratio [total-to-total], dmls
C	PRESS(i)	REAL	Working Fluid Pressure at Statepoint "i", psia
C	PREXP	REAL	Prandtl No. Exponent Used in Heat Transfer Calcs., dmls
C	PRNDTL	REAL	Current Heat Transfer "Prandtl No.", dmls
C	PWRFCTR	REAL	Load Power Factor at Generator Terminals [lagging]
C	QHSA	REAL	Heat Source Gross Thermal Power, kWt
C	QLOSSi	REAL	Thermal Loss for Section Ending at Statepoint "i", kWt
C	RADHC	REAL	Axial Compressor Flowpath Hub Radius, inches
C	RAD1C	REAL	Axial Compressor Flowpath Inlet Tip Radius, inches
C	RADNC	REAL	Axial Compressor Flowpath Discharge Tip Radius, inches
C	RADHT	REAL	Axial Turbine Flowpath Hub Radius, inches
C	RAD1T	REAL	Axial Turbine Flowpath Inlet Tip Radius, inches
C	RADNT	REAL	Axial Turbine Flowpath Discharge Tip Radius, inches
C	RECH	REAL	Recuperator Core Height, inches
C	RECLC	REAL	Recuperator Core Length, inches
C	RECLOA	REAL	Recuperator Overall Length, inches
C	RECW	REAL	Recuperator Core Width, inches
C	RHOFINS	REAL	Recup. Fins Density; s=L or H for Location, 1bm/cu in
C	ROPLAT	REAL	Recuperator Splitter Plate Density, 1bm/cu in
C	ROBRAZ	REAL	Recuperator Braze Density, 1bm/cu in
C	SIGMAS	REAL	Recup. Flow:Frontal Area Ratio; s=L or H for Loc., dmls
C	SPACES	REAL	Recuperator Plate Spacing; s=L or H for Location, in.
C	SPEED	REAL	Generator Design Rotating Speed, rpm
C	TBRAZE	REAL	Recuperator Braze Thickness [4/pitch], inches
C	TEMP(i)	REAL	Working Fluid Temperature at Statepoint "i", deg R
C	TFINs	REAL	Recuperator Fins Thickness; s=L or H for Location, in.
C	TINCLNT	REAL	Specified Generator Coolant Inlet Temperature, deg R
C	TIPSPDG	REAL	Generator Rotor Surface Speed, ft/sec [limit is 700]
C	TITLER(8)	CH*10	Description of Recuperator Core Matrix Geometry
C	TOUTCLNT	REAL	Specified Generator Coolant Outlet Temperature, deg R
C	TPLATE	REAL	Recuperator Splitter Plate Thickness, inches
C	TURBDIA	REAL	Radial Turbine Wheel Tip Diameter, inches
C	TURBSS	REAL	Turbine Specific Speed
C	TURBTYPE	CH*10	Turbine Type, either 'AXIAL' or 'RADIAL'
C	TVAR(18)	CH*10	Variable Names [as converted in SUBROUTINE SETUP]
C	UAMH	REAL	UA [heat transfer] DesignMargin for Recuperator High Pressure Side, dmls
C	UAML	REAL	UA [heat transfer] DesignMargin for Recuperator Low Pressure Side, dmls
C	UTIPC	REAL	Compressor [maximum for axials] Tip Speed, ft/sec
C	UTIPT	REAL	Turbine [maximum for axials] Tip Speed, ft/sec

C	VGAS(6)	REAL	Velocity of the gas in individual duct segments, ft/sec
C	VOLTAGE	REAL	Desired Generator Output Voltage, 3Ph, line-line, RMS
C	WARNINGC	CH*64	Compressor Subroutine Diagnostic Warning Message
C	WARNINGF	CH*64	Fluid Properties Subroutine Diagnostic Warning Message
C	WARNINGG	CH*64	Generator Subroutine Diagnostic Warning Message
C	WARNINGR	CH*64	Recuperator Subroutine Diagnostic Warning Message
C	WARNINGT	CH*64	Turbine Subroutine Diagnostic Warning Message
C	WCLNT	REAL	Calculated Coolant Flow Rate, lbm/sec
C	WGTDUCT(6)	REAL	Mass of the individual duct segments with MFI, 1bm
C	WDCTTOT	REAL	Total of Duct Masses, 1bm
C	WGTHRA	REAL	Heat Rejection Assembly Mass, 1bm
C	WGTHSA	REAL	Heat Source Assembly Mass, 1bm
C	WGTREC	REAL	Recuperator Assembly Mass, 1bm
C	WGTTAC	REAL	Turbo-alternator Assembly Mass, 1bm
C	WGTTOT	REAL	Total PCU Mass, 1bm
C	WINDAGE	REAL	Generator Main Gap Windage Loss, kWs
C	XBLC	REAL	Bleed Fraction of Flow(1) Returned to Statepoint #2 [comp wheel tip], fraction
C	XBLT	REAL	Bleed Fraction of Flow(1) Returned to Statepoint #13 [turbine stage exit], fraction

C \*\*\*\*\* Statepoint Definition \*\*\*\*\*

C Statepoint                          Location

C Note: Total [stagnation] conditions except as noted.

C # 1 Compressor Inlet

C Note: Statepoints # 2, 3, 4 are relevant with radial compressors only

C # 2 Compressor Rotor Tip [static pressure]

C # 3 Compressor Diffuser Inlet

C # 4 Compressor Diffuser Exit [comp-end bleed flow added]

C # 5 Compressor Stage Exit [comp-end bleed flow removed]

C # 6 Recuperator High Pressure Flow Inlet

C # 7 Recuperator High Pressure Flow Discharge

C # 8 HSA Inlet

C # 9 HSA Discharge

C # 10 Turbine Inlet

C Note: Statepoints # 11 is relevant with radial turbines only

C # 11 Turbine Nozzel Exit

C # 12 Turbine Discharge

C # 13 Recuperator Low Pressure Flow Inlet [turb-end bleed flow added]

C # 14 Recuperator Low Pressure Flow Discharge

C # 15 HRA Inlet

C # 16 HRA Discharge

C # 17 Compressor Inlet Duct Exit [= to Statepoint #1]

C \*\*\*\* Define the efficiencies, masses, sizes of the resulting PCU components  
C \*\*\*\* This subroutine sizes a single PCU at the specified power level.  
C \*\*\*\* Initialize certain cycle parameters with appropriate starting values.

PRESS(1)=100.  
EFFC=.85  
EFFT=.9  
EFFA=.96  
MOLWT=VAR(3)  
WINDAGE=0.  
BRGLOSS=0.  
COOLING=0.

C \*\*\*\* Call SUBROUTINE SETUP to load independent cycle parameters desired.

CALL SETUP

C \*\*\*\* Load tanspoert parameters for working fluid; binary mixture or  
C \*\*\*\* single Noble gas.

CALL GASPROP

C \*\*\*\* Compute Brayton statepoints, efficiencies, losses

CALL STATEPT

C \*\*\*\*\* Error trap inserted following telecon with LeRC, 7-9-93 \*\*

IF (ERRORC.EQ.  
& Negative compressor inlet pressure predicted in STATEPT') RETURN

C \*\*\*\*\*

C \*\*\*\* Compute Intermediate Heat Exchanger (IHX) size and mass.  
C \*\*\*\* The IHX code, "IHXSHEL", size the IHX. The code "IHXSHEL"  
C iterates on "PRATIO" (tube pitch ratio) until the desired  
C gas side pressure drop is achieved.

C \*\*\*\* Input data required for IHXSHEL

C data file IHXflg - 1=Lithium, 2=NaK  
C data file UEST1 - Initial value for U overall (Btu/hr-ft<sup>2</sup>-R)  
C data file THIN1 - Hot side inlet temperature, R  
C data file THOUT1 - Hot side outlet temperature, R  
C PHOT1 - Gas side inlet pressure, psia  
C TCIN1 - Cold (gas) side inlet temperature, R  
C TCOOUT1 - Cold (gas) side outlet temperture, R  
C WDOTS1 - Shell (gas) side fluid flow rate, lbm/s  
C AMWS1 - Molecular weight of He/Xe mixture, lbm/lbmole  
C data file TINS1 - Insulation thickness, in  
C data file DENINS1 - Bulk density of insulation, lbm/ft<sup>3</sup>

```

C data file DENSSH1 - Density of shell, 1bm/ft3
C data file DTUBE1 - Inside diameter of heat exchanger tubes, in
C      PRATIO - Tube pitch ratio (ITERATED TO GET DESIRED DP)
C data file TTUBE - Tube wall thickness, in
C data file ANPLATES - Number of support plates
C      WDOTT - Tube side mass flow rate, 1bm/s
C data file AKTUBE - Tube wall thermal conductivity, Btu/hr-ft-R
C      QDOT - Heat rate or duty, Btu/hr

C note - the items read in via the data file were converted to
c      engineering after being read in. The data contained in the
c      data file are in SI units.

C The following is the cycle data required to size the heat exchanger
PHOT1=PRESS(8)
TCIN1=TEMP(8)
TCOUT1=TEMP(9)
WDOTS1=FLOW(8)
AMWS1=MOLWT
QDOT1=FLOW(8)*(4.97/MOLWT)*(TEMP(9)-TEMP(8))*3600.

C **** corrected Taverage computation, 7-30-93 ****
TAVERAGE=(THIN1+THOUT1)/2.0
C ****
CALL XLITHP(TAVERAGE,BBRHO,CPHOT,BBVIS,BBTK)
IF (IHXFLG .EQ. 2) THEN
CALL XNAKPR(TAVERAGE,BBRHO,CPHOT,BBVIS,BBTK)
END IF
WDOTT1=QDOT1/((CPHOT)*(THIN1-THOUT1)*3600.)

C Iterate on the pitch ratio until the desired gas side pressure drop
C is achieved
PRATIO=2.0

2 CALL IHXSHEL(IHXflg,UEST1,THIN1,THOUT1,PHOT1,TCIN1,TCOUT1,WDOTS1,
&AMWS1,TINS1,DENINS1,DENSSH1,DTUBE1,PRATIO,TTUBE1,ANPLATES1,
&WDOTT1,AKTUBE1,QDOT1,DPSHELL1,ANTUBES1,DPTUBE1,DOTL21,ALSHEL1,
&AMSHELL1,AMPLATES1,AMTUBES1,AMINSUL1,AMHEADS1,AMSTRT1,ANETMASS1,
&XMNHEX1,HSHELL1,AFRIC1,UNEW1,RETUBE1,THC1,AMTSHT1)

DP=PRESS(8)-PRESS(9)
DELTA=DPSHELL1-DP
ERROR=ABS(DELTA/DP)
IF (ERROR.GT..05) THEN
PRATIO=PRATIO*(1+((DELTA/DP)*.1))
WDOTS1=FLOW(8)
QDOT1=FLOW(8)*(4.97/MOLWT)*(TEMP(9)-TEMP(8))*3600.

```

```

TAVERAGE=(THIN1+THOUT1)/2.0
CALL XLITHP(TAVERAGE,BBRHO,CPHOT,BBVIS,BBTK)
IF (IHXFLG .EQ. 2) THEN
CALL XNAKPR(TAVERAGE,BBRHO,CPHOT,BBVIS,BBTK)
END IF
WDOTT1=QDOT1/((CPHOT)*(THIN1-THOUT1)*3600.)
GO TO 2
END IF

```

C \*\* Output from the heat exchanger sizing

```

c DPSHELL1 - shell side pressure drop, psi
c ANTUBES1 - number of heat exchanger tubes
c DPTUBE1 - tube side pressure drop, psi
c DOTL21 - diameter of the heat exchanger shell, in
c ALSHEL1 - length of the heat exchanger shell, in
c AMSHELL1 - mass of the shell, 1bm
c AMPLATES1- mass of the plates, 1bm
c AMTUBES1 - mass of the tubes, 1bm
c AMINSUL1 - mass of the insulation, 1bm
c AMHEADS1 - mass of the heads, 1bm
c AMSTRT1 - mass of structure and brackets, 1bm
c ANETMASS1- net mass of the shell and tube unit, 1bm
c XMNHEX1 - mass of the tubeside fluid, 1bm
c HSHELL1 - shell side h
c AFRIC1 - friction factor
c UNEW1 - overall U, Btu/hr-ft2-R
c RETUBE1 - tubeside Reynolds number
c THC1 - tubeside h, Btu/hr-ft2-R
c AMTSHT1 - mass of the tubesheets, 1bm

```

C \*\*\*\*

C \*\*\* At this point the code will call the heat rejection module.

C The overall integrator of the code must:

- c 1. Provide appropriate cycle data required by the heat rejection module
  - c 2. Provide any geometry related information, if any, for the heat rejection module.
  - c 3. Provide the required iteration routine to get the desired gas side pressure drop on the heat rejection gas cooler.
- c The heat rejection gas cooler routine is identical to the IHX routine. Iterating on the tube pitch ratio, as done
- c with the IHX, will be required.

C \*\*\*\*

C \*\*\*\* Evaluate ducting size and mass.

```

ERR1=' '
ERR2=' '
ERR3=' '

```

```
ERR4=' '
ERR5=' '
ERR6=' '
ERR7=' '
ERR8=' '
ERR9=' '
ERR10=' '
ERR11=' '
ERR12=' '

IDUCT = 1
PIN = PRESS(5)
POUT = PRESS(6)
TIN = TEMP(5)
MDOT = FLOW(5)
MW = MOLWT

CALL DUCTING

IDUCT = 2
IF (EFFR .GT. 0.0) THEN
PIN = PRESS(7)
POUT = PRESS(8)
TIN = TEMP(7)
MDOT = FLOW(7)
MW = MOLWT
CALL DUCTING
END IF

IDUCT = 3
PIN = PRESS(9)
POUT = PRESS(10)
TIN = TEMP(9)
MDOT = FLOW(9)
MW = MOLWT
CALL DUCTING

IDUCT = 4
PIN = PRESS(12)
POUT = PRESS(13)
TIN = TEMP(12)
MDOT = FLOW(12)
MW = MOLWT
CALL DUCTING

IDUCT = 5
IF (EFFR .GT. 0.0) THEN
PIN = PRESS(14)
POUT = PRESS(15)
TIN = TEMP(14)
MDOT = FLOW(14)
MW = MOLWT
```

```
CALL DUCTING
END IF

IDUCT = 6
PIN = PRESS(16)
POUT = PRESS(17)
TIN = TEMP(16)
MDOT = FLOW(16)
MW = MOLWT
CALL DUCTING
```

C \*\*\*\* Evaluate recuperator heat transfer, pressure loss, size and mass.

```
CALL RECSIZE
```

C \*\*\*\* Compute the mass of the TAC and remaining system components.

```
CALL TAC
```

C \*\*\*\* Sum the fout to six duct masses (dependent on recuperation scheme).

```
WDCTTOT=0.
```

C \*\*\*\* Compute total mass for one PCU (single TAC/recup/hxda).

```
WGTTOT=WGTHRA+WDCTTOT+WGTTAC+WGTRREC+WGTHSA
IF (IPRINT.GT.0) THEN
CALL OUTPUT
CLOSE (61)
END IF
RETURN
END
```

#### SUBROUTINE SETUP

C \*\*\*\* SETUP is checked-out and final on October 6, 1992

```
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
REAL JCON
```

```
CHARACTER*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18),
& COMPTYPE,TURBTYPE,CLNTTYPE
CHARACTER*20 GENTYPE,INTTYPE
```

```
COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
& RADNC,ANSQC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,
& RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT
COMMON/OPTIM/ VAR(18),TVAR,IPRINT
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
```

```
COMMON/TITLE/ CYCDES,TITLER
COMMON/HSACALCS/ QHSA,WGTHSA,DP9
COMMON/LOSS/ QLOSS8,QLOSS9,QLOSS10,QLOSS13,FCTQ8,FCTQ10,FCTQ13
COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLO,
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,
& DPOP16,DPOP17,NETSP,GROSSEP,EFFCYCLE,WINDAGE
COMMON/GASBRG/BRGLOSS
COMMON/RECUP/ DPOPREC,UAML,UAMH,DPML,DPMH,
& PREXP,NFINL,TFINL,SPACEL,LEQL,ROFINL,KFINL,NFINH,TFINH,SPACEH,
& LEQH,ROFINH,KFINH,TPLATE,ROPLAT,TBRAZE,ROBRAZ,
& SIGMAL,ALPHAL,AFINL,APlATL,HDL,SIGMAH,ALPHAH,AFINH,APlATH,HDH,
& PITCH,WGTREC
```

```
C      COMMON/RADTR/ PREXPM,TITLER,EPsiLON,ETAFINR,HPRED,RADSM,NFINM,
COMMON/CONST/PI,RU,GO,JCON,RZERO
```

```
C **** Load desired design parameters form the input variable array.
```

```
TEMP(1)=VAR(1)
PRC=VAR(2)
MOLWT=VAR(3)
EFFR=VAR(4)
COMPSS=VAR(5)
SPEED=VAR(6)
DPOP6=VAR(7)
DPOPREC=VAR(8)
DPOP8=VAR(9)
DPOP9=VAR(10)
DPOP10=VAR(11)
DPOP13=VAR(12)
DPOP15=VAR(13)
DPOP16=VAR(14)
DPOP17=VAR(15)
```

```
C **** If EFFR = 0. the cycle is unrecuperated, if > 0. is recuperated.
```

```
IF (EFFR.GT.0.) THEN
DPOP7=1.-(1.-DPOPREC)**.4
DPOP14=1.-(1.-DPOPREC)**.6
ELSE
```

```
C **** Without recuperator present, duct losses are defined by DPOP6,
C **** DPOP10, DPOP13, and DPOP17.
```

```
DPOP7=0.
DPOP14=0.
DPOP8=0.
VAR(9)=0.
DPOP15=0.
VAR(13)=0.
```

END IF

RETURN  
END

SUBROUTINE OUTPUT

C \*\*\*\* OUTPUT is checked-out and final on October 6, 1992  
IMPLICIT REAL (A-H)  
IMPLICIT INTEGER (I-J)  
IMPLICIT REAL (K-Z)

REAL JCON  
CHARACTER\*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18),  
& COMPTYPE,TURBTYPE,CLNTTYPE  
CHARACTER\*20 GENTYPE,INTTYPE  
CHARACTER\*64 ERRORT,ERRORC,ERRORR,ERRORG,ERRORF,WARNIGT,WARNIGC,  
& WARNINGR,WARNIGG  
CHARACTER\*60 ERR1, ERR2, ERR3, ERR4, ERR5, ERR6, ERR7, ERR8,  
& ERR9, ERR10, ERR11, ERR12  
  
COMMON/DIAGNOS/ERRORT,ERRORC,ERRORR,ERRORG,ERRORF,WARNIGT,  
& WARNIGC,WARNINGR,WARNIGG  
COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE  
COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,  
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,  
& CPCLN, LIFETIME  
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,  
& RADNC,ANSQC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,  
& RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT  
COMMON/OPTIM/ VAR(18),TVAR,IPRINT  
COMMON/OUTP/WGTTOT,WGTTAC,RECLC,RECLOA,RECH,RECW  
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDL  
COMMON/TITLE/ CYCDES,TITLER  
  
COMMON /DUCTOUT/ WGTDUCT(6), DIAM(6), LGTH(6), VGAS(6)  
COMMON /DUCTERR/ ERR1, ERR2, ERR3, ERR4, ERR5, ERR6, ERR7, ERR8,  
& ERR9, ERR10, ERR11, ERR12  
COMMON/CONVERT/ XTEMP(17), XPRESS(17), XFLOW(17), XDIAM(6),  
& XLGTH(6), XVGAS(6), XWGTDUCT(6)  
COMMON/HSACALCS/ QHSA,WGTHSA,DP9  
COMMON/HRACALCS/ QHRA,WGTHRA,DP16  
COMMON/LOSS/ QLOSS8,QLOSS9,QLOSS10,QLOSS13,FCTQ8,FCTQ10,FCTQ13  
COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLO,  
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCR,  
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,  
& DPOP16,DPOP17,NETSP,GROSSEP,EFFCYCLE,WINDAGE  
COMMON/GASBRG/BRGLOSS  
COMMON/RECUP/ DPOPREC,UAML,UAMH,DPMI,DPMH,  
& PREXP,NFINL,TFINL,SPACEI,LEQL,ROFINL,KFINL,NFINH,TFINH,SPACEH,  
& LEQH,ROFINH,KFINH,TPLATE,ROPLAT,TBRAZE,ROBRAZ,  
& SIGMAL,ALPHAL,AFINL,APLATL,HDL,SIGMAH,ALPHAH,AFINH,APLATH,HDH,

```

& PITCH,WGTREC
COMMON/MASSES/MASSCMP,MASSTRB,MWHLIC,MWHLT,MSFTC,MSFTT,MCASEC,
& MCASET,MDIAC,MDIAT
COMMON/IHX/ DPSHELL1,ANTUBES1,DPTUBE1,DOTL21,ALSHEL1,AMSHELL1,
& AMPLATES1,AMTUBES1,AMINSUL1,AMHEADS1,AMSTRT1,ANETMASS1,XMNHEX1,
& HSHELL1,AFRIC1,UNEW1,RETUBE1,THC1,AMTSHT1,PRATIO, IHXFLG, UEST1,
& THIN1, THOUT1, TTUBE1, ANPLATES1,AKTUBE1, TINS1, DENINS1,
& DENSSH1, DTUBE1

COMMON/CONST/PI,RU,G0,JCON,RZERO

      WRITE(61,9)
9 FORMAT(/,',',/)
      WRITE(61,10) (CYCDES(I),I=1,8)
      WRITE(61,10) (TITLER(I),I=1,8)
10 FORMAT (1X,8A10)
      WRITE (61,20)

C *** convert R to K
var(1)=var(1)/1.8

20 FORMAT (' *** PARAMETRIC DATA ****')
      WRITE (61,30) (TVAR(I),VAR(I),I=1,15)
30 FORMAT (3(4X,A8,E12.5))
      WRITE (61,40)
40 FORMAT (' *** MASS and EFFICIENCY DATA ****')
      WRITE (61,40)

C *** convert masses from 1bm to kg

XWGTTAC=WGTTAC*0.45359
XWGTREC=WGTREC*0.45359
XWDCTTOT=WDCTTOT*0.45359
XMASSCMP=MASSCMP*0.45359
XMASSTRB=MASSTRB*0.45359
XMASSGEN=MASSGEN*0.45359
XWGTTOT=WGTTOT*0.45359
XWDCTTOT=0
DO 49 I=1,6
  XWDCTTOT=XWDCTTOT+(WGTDUCT(I)*.45359)
49 CONTINUE


```

C\*\*\*\*\* ADD COMPUTATION FOR POWER CONVERSION SYSTEM MASS 7-30-93 \*\*\*\*\*

PCSTOT=XWGTTOT+XWDCTTOT

C\*\*\*\*\*

WRITE (61,50) XWGTTAC,XWGTREC,XWDCTTOT,
& XMASSCMP,XMASSTRB,XMASSGEN,PCSTOT,EFFCYCLE,
& EFFC,EFFT,EFFA,BETA,GROSSEP

```

50 FORMAT (' TAC MASS, KG ',F10.0,',      REC MASS, KG ',F10.0,/,
&          ' DCTNG MASS, KG ',F10.0,/,      TRB MASS, KG ',F10.0,/,
&          ' CMP MASS, KG ',F10.0,/,      PCS MASS, KG ',F10.0,/,
&          ' GEN MASS, KG ',F10.0,/,      ETA COMPR  ',F10.5,/,
&          ' CYC EFF      ',F10.5,/,      ETA ALT    ',F10.5,/,
&          ' ETA TURB     ',F10.5,/,      GROSS EP kWe ',F10.1)

C *** convert parameters from engineering units for printout

XRADHC=RADHC*2.54
XRAD1C=RAD1C*2.54
XRADNC=RADNC*2.54
XMASSCMP=MASSCMP*.45359
XCOMPDIA=COMPDIA*2.54
XUTIPC=UTIPC*.3048
XRADHT=RADHT*2.54
XRAD1T=RAD1T*2.54
XRADNT=RADNT*2.54
XMASSTRB=MASSTRB*.45359
XTURBDIA=TURBDIA*2.54
XUTIPT=UTIPT*.3048
XDGENRTR=DGENRTR*2.54
XDGENSTR=DGENSTR*2.54
XLGENTOT=LGENTOT*2.54
XMASSGEN=MASSGEN*.45359
XTIPSPDG=TIPSPDG*.3048
XTINCLNT=TINCLNT/1.8
XTOUTCLNT=TOUTCLNT/1.8
XWCLNT=WCLNT*.45359

      WRITE (61,60)
60 FORMAT ('/ *** DIMENSIONAL DATA ****
&*****')
      IF (COMPTYPE.EQ.'AXIAL') THEN
      WRITE (61,65) ISTGC,XRADHC,XRAD1C,XRADNC,ANSQC,ANSQCL,COMPSS,
&XMASSCMP
      ELSE
      WRITE (61,80) XCOMPDIA,XUTIPC,COMPSS,XMASSCMP
      END IF
      IF (WARNINGC.NE.' ') WRITE (61,67) WARNINGC
      IF (ERRORC.NE.' ') WRITE (61,68) ERRORC
      IF (TURBTYPE.EQ.'AXIAL') THEN
      WRITE (61,66) ISTGT,XRADHT,XRAD1T,XRADNT,ANSQT,ANSQTL,TURBSS,
&XMASSTRB
      ELSE
      WRITE (61,81) XTURBDIA,XUTIPT,TURBSS,XMASSTRB
      END IF
      IF (WARNINGT.NE.' ') WRITE (61,67) WARNINGT
      IF (ERRORT.NE.' ') WRITE (61,68) ERRORT

```

```

65 FORMAT(' ***** AXIAL COMPRESSOR GEOMETRY',/,
& ' NO STAGES   ',I5,5X,' HUB RAD, cm ',F10.3,
& ' TIP R INL,cm ',F10.3/,'
& ' TIP R DCH,cm ',F10.3,' AN^2C      'E10.3,
& ' AN^2LIMIT   ',E10.3/,'
& ' SPEC SPD    ',F10.1,' MASS, kg     'F10.1)

66 FORMAT(' ***** AXIAL TURBINE GEOMETRY',/,
& ' NO STAGES   ',I5,5X,' HUB RAD, cm ',F10.3,
& ' TIP R INL,cm ',F10.3/,'
& ' TIP R DCH,cm ',F10.3,' AN^2T      'E10.3,
& ' AN^2LIMIT   ',E10.3/,'
& ' SPEC SPD    ',F10.1,' MASS, kg     'F10.1)

67 FORMAT (1X,'WARNING = ',A64)
68 FORMAT (1X,' ERROR = ',A64)

c ***** Add frequency to output, 7-30-93 *****
FREQNCY=VAR(6)/60.0

c *****
WRITE (61,83) XDGENRTR,XDGENSTR,XLGENTOT,GENASP,XMASSGEN,
&XTIPSPDG,KVA,PWRFCCTR,VOLTAGE,FREQNCY,XTINCLNT,XTOUTCLNT,XWCLNT
83 FORMAT (' ***** GENERATOR GEOMETRY AND PARAMETERS'/
& ' Rotor OD, cm ',F10.3,' Stator OD,cm ',F10.3,
& ' Length, cm   ',F10.3/,'
& ' Rotor L/D    ',F10.2,' Mass, kg     ',F10.2,
& ' Tip Spd m/s  ',F10.0/,'
& ' KVA          ',F10.1,' Pwr Factr   ',F10.3,
& ' Volts,l-l    ',F10.0/,'
& ' Frequency, Hz ',F10.3/,'
& ' CInt Tin, K  ',F10.0,' CInt Tout, K ',F10.0,
& ' CInt Flow,kg/s',F10.2)

WRITE (61,69) GENTYPE,INTTYPE,CLNTTYPE
IF (WARNINGG.NE.' ') WRITE (61,67) WARNINGG
IF (ERRORG.NE.' ') WRITE (61,68) ERRORG
69 FORMAT (' TYPE= ',A20,' INTERFACE= ',A20,/,'
& ' COOLANT= ',A10)

80 FORMAT (' ***** RADIAL COMPRESSOR GEOMETRY'/
& ' COMP DIA, cm  ',F10.3,' TIP SPD.,m/s ',F10.0,
& ' SPEC SP      ',F10.3/
& ' COMP MASS, kg ',F10.1)

81 FORMAT (' ***** RADIAL TURBINE GEOMETRY'/
& ' TURB DIA, cm  ',F10.3,' TIP SPD., m/s ',F10.0,
& ' SPEC SP      ',F10.3/

```

```

& ' TURB MASS, kg ',F10.1)

      WRITE (61,90)
90 FORMAT (' *** THERMODYNAMIC DATA ****
&*****')
C ** convert to K, kPa, kg/s

      DO 95 I=1,17
      XTEMP(I)=TEMP(I)/1.8
      XPRESS(I)=PRESS(I)*6.8948
      XFLOW(I)=FLOW(I)*0.45359
95 CONTINUE

      WRITE (61,100) (XTEMP(I),I=1,17)
      WRITE (61,110) (XPRESS(I),I=1,17)
      WRITE (61,120) (PR(I),I=1,17)
      WRITE (61,130) (XFLOW(I),I=1,17)
100 FORMAT (' TEMPERATURES, K ',6F10.2)
110 FORMAT (' PRESSURES, KPA ',6F10.2)
120 FORMAT (' PRESS RATIOS ',6F10.6)
130 FORMAT (' FLOW RATES, KG/S ',6F10.2)

      WRITE (61,140)
140 FORMAT (' *** PARASITIC LOSSES ****
&*****')
      WRITE (61,150) QLOSS9,QLOSS8,QLOSS10,QLOSS13
150 FORMAT (' THERMAL LOSSES [kW]- HSA ',F10.2,
& ' , THTML 8 ',F10.2,/,,
& ' , THRML 10 ',F10.2,
& ' , THRML 13 ',F10.2)
      WRITE (61,152) BRGLOSS,WINDAGE
152 FORMAT (' SHAFT LOSSES [kW]- TOT BRGS',F10.2,' WINDAGE ',F10.2)

      WRITE (61,155)
155 FORMAT ('//,***** Ducting Configuration *****',/)

C ** convert to cm, m/s, kg

      DO 157 I=1,6
      XDIAM(I)=DIAM(I)*2.54
      XLGTH(I)=LGTH(I)*2.54
      XVGAS(I)=VGAS(I)*0.3048
      XWGTDUCT(I)=WGTDUCT(I)*0.45359
157 CONTINUE

      WRITE (61,164) (I,XDIAM(I),XLGTH(I), I=1,6)

      WRITE (61,163)
163 FORMAT (' ',/)


```

```

      WRITE (61,165) (I,XVGAS(I),XWGTDUCT(I), I=1,6)

164 FORMAT (T1,'Duct',I2,' diameter, cm =',F10.5,' Length, cm=',,
&F10.5)
165 FORMAT (T1,'Duct',I2,' velocity, m/s=',F10.5,' Weight, kg=',,
&F10.5)

      IF (ERR1.NE.' ') WRITE (61,166) ERR1
      IF (ERR2.NE.' ') WRITE (61,166) ERR2
      IF (ERR3.NE.' ') WRITE (61,166) ERR3
      IF (ERR4.NE.' ') WRITE (61,166) ERR4
      IF (ERR5.NE.' ') WRITE (61,166) ERR5
      IF (ERR6.NE.' ') WRITE (61,166) ERR6
      IF (ERR7.NE.' ') WRITE (61,166) ERR7
      IF (ERR8.NE.' ') WRITE (61,166) ERR8
      IF (ERR9.NE.' ') WRITE (61,166) ERR9
      IF (ERR10.NE.' ') WRITE (61,166) ERR10
      IF (ERR11.NE.' ') WRITE (61,166) ERR11
      IF (ERR12.NE.' ') WRITE (61,166) ERR12

166 FORMAT (1X, A60)

C *** Convert IHX parameters from engineering units for printout
XDPSHELL1 = DPSHELL1*6.8946
XDPTUBE1 = DPTUBE1*6.8948
XDOTL21 = DOTL21*2.54
XALSHEL1 = ALSHEL1*2.54
XAMSHELL1= AMSHELL1*.45359
XAMPLATES1 = AMPLATES1*.45359
XAMTUBES1 = AMTUBES1*.45359
XAMINSUL1 = AMINSUL1*.45359
XAMHEADS1 = AMHEADS1*.45359
XAMSTR1 = AMSTR1*.45359
XANETMASS1 = ANETMASS1*.45359
XXMNHEX1 = XMNHEX1*.45359
XHSHELL1 = HSHELL1*1054.4*1.8/(3600.*.3048*.3048)
XUNEW1 = UNEW1*1054.4*1.8/(3600.*.3048*.3048)
XTHC1 = THC1*1054.4*1.8/(3600.*.3048*.3048)
XAMTSHT1 = AMTSHT1*.45359

C ***** ADD COMPUTATION FOR WET IHX MASS, 7-30-93 *****
      XIHXWET=XANETMASS1+XXMNHEX1

C ****
      WRITE (61,200) XDPSHELL1,ANTUBES1,XDPTUBE1,XDOTL21,XALSHEL1,
&XAMSHELL1,XAMPLATES1,XAMTUBES1,XAMINSUL1,XAMHEADS1,XAMTSHT1,
&XAMSTR1,XANETMASS1,XXMNHEX1,XHSHELL1,AFRIC1,XUNEW1,RETUBE1,
&XTHC1,PRATIO

```

```

C ** Output from the heat exchanger sizing
200 FORMAT (/,<4X,'IHX Design Parameters',/,,
& ' XDPSHELL1 - shell side pressure drop, kPa ',F10.3,/,
& ' ANTUBES1 - number of heat exchanger tubes ',F10.0,/,
& ' XDPTUBE1 - tube side pressure drop, kPa ',F10.3,/,
& ' XDOTL21 - diameter of the shell, cm ',F10.1,/,
& ' XALSHEL1 - length of the shell, cm ',F10.1,/,
& ' XAMSHELL1 - mass of the shell, kg ',F10.1,/,
& ' XAMPLATES1 - mass of the plates, kg ',F10.1,/,
& ' XAMTUBES1 - mass of the tubes, kg ',F10.1,/,
& ' XAMINSULL - mass of the insulation, kg ',F10.1,/,
& ' XAMHEADS1 - mass of the heads, kg ',F10.1,/,
& ' XAMTSHT1 - mass of the tubesheets, kg ',F10.1,/,
& ' XAMSTR1 - structure and brackets, kg ',F10.1,/,
& ' XANETMASS1 - net shell and tube unit, kg ',F10.1,/,
& ' XXMNHEX1 - mass of tubeside fluid, kg ',F10.1,/,
& ' XHSHELL1 - shell side h, W/m2-K ',F10.1,/,
& ' AFRIC1 - friction factor ',F10.3,/,
& ' XUNEW1 - overall U, W/m2-K ',F10.5,/,
& ' RETUBE1 - tubeside Reynolds number ',F10.3,/,
& ' XTHC1 - tubeside h, W/m2-K ',F10.1,/,
& ' PRATIO - tube pitch ratio ',F10.1,/)

```

C \*\*\* ADD A MASS SUMMARY TABLE, 7-30-93 \*\*\*\*\*

WRITE(61,210) XWGTTAC, XWGTREC, XWDCTTOT, PCSTOT, XIHXWET

```

210 FORMAT (//,
& ' SUMMARY MASS TABLE',//,
& ' TAC Mass, kg ', f10.3,/,
& ' Recuperator Mass, kg ', f10.3,/,
& ' Ducting mass, kg ', f10.3,/,
& ' Power Conversion System, kg ', f10.3,/,//,
& ' Intermediate heat exchanger (wet), kg ', f10.3,/)

```

C \*\*\*\*\*

END

SUBROUTINE STATEPT

C \*\*\*\* STATEPT is checked-out and final on October 6, 1992  
\$ DEBUG

IMPLICIT REAL (A-H)  
IMPLICIT INTEGER (I-J)  
IMPLICIT REAL (K-Z)  
REAL JCON

CHARACTER\*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18),  
& COMPTYPE,TURBTYPE,CLNTTYPE  
CHARACTER\*20 GENTYPE,INTTYPE

CHARACTER\*64 ERRORT, ERRORC, ERRORR, ERRORG, ERRORF, WARNINGT, WARNINGC,  
& WARNINGR, WARNINGG

COMMON/DIAGNOS/ERRORT, ERRORC, ERRORR, ERRORG, ERRORF, WARNINGT,  
& WARNINGC, WARNINGR, WARNINGG  
COMMON/CONFIG/COMPTYPE, TURBTYPE, GENTYPE, INTTYPE, CLNTTYPE  
COMMON/ALTERNTR/DGENRTR, DGENSTR, LGENTOT, MASSGEN, TIPSPDG, COE,  
& ETAGEN, COOLING, WCLNT, VOLTAGE, KVA, GENASP, TINCLNT, TOUTCLNT,  
& CPCLNT, LIFETIME  
COMMON/AERODYN/ETACOMP, COMPDIA, UTIPC, RADHC, RAD1C,  
& RADNC, ANSQC, ANSQCL, COMPSS, ISTGC, ETATURB, TURBDIA, UTIPT,  
& RADHT, RAD1T, RADNT, ANSQT, ANSQLT, TURBSS, ISTGT  
COMMON/OPTIM/ VAR(18), TVAR, IPRINT  
COMMON/FLUID/ GAS, GAMMA, MOLWT, CP, PRNDL  
COMMON/TITLE/ CYCDES, TITLER  
COMMON/HSACALCS/ QHSA, WGTHSA, DP9  
COMMON/HRACALCS/ QHRA, WGTHRA, DP16  
COMMON/LOSS/ QLOSS8, QLOSS9, QLOSS10, QLOSS13, FCTQ8, FCTQ10, FCTQ13  
COMMON/CYCLE/TEMP(17), PRESS(17), FLOW(17), PR(17), BETA, COMFLO,  
& PRC, EFFC, EFFT, EFFR, EFFA, XBLC, XBLT, SPEED, POWER, PWRFCTR,  
& DPOP6, DPOP7, DPOP8, DPOP9, DPOP10, DPOP13, DPOP14, DPOP15,  
& DPOP16, DPOP17, NETSP, GROSSEP, EFFCYCLE, WINDAGE  
COMMON/GASBRG/BRGLOSS  
COMMON/CONST/PI, RU, GO, JCON, RZERO

ITER=0

C \*\*\*\* Initialize all pressure ratios to 1 [pressure losses = 0]

DO 5, I=1,17  
5 PR(I)=1.

C \*\*\*\* Compute PR(i)'s not = 1 [flows with assigned/calculated losses]

PR(6) = 1.-DPOP6  
PR(7) = 1.-DPOP7  
PR(8) = 1.-DPOP8

C \*\*\*\* DPOP(9) [Heat Source delta P/P} is supplied by Rocketdyne  
C and/or NASA and is contained within the statepoint iteration  
C \*\*\*\* loop below.

PR(10) = 1.-DPOP10  
PR(13) = 1.-DPOP13  
PR(14) = 1.-DPOP14  
PR(15) = 1.-DPOP15

C \*\*\*\* DPOP(9) [Heat Rejection delta P/P} is supplied by Rocketdyne  
C \*\*\*\* and is contained within the statepoint iteration loop below.

PR(16) = 1.-DPOP16

PR(17) = 1.-DPOP17

C \*\*\*\* Make an initial pass through GENSIZE to validate or reestablish the  
C rotating speed value. If generator tip speed exceeds 700 fps a new  
C \*\*\* speed will be defined.

CALL GENSIZE  
EFFA=ETAGEN

C \*\*\*\* Begin Power Iteration Loop Begins

10 CONTINUE  
PR(9)=1.-DPOP9  
PR(16)=1.-DPOP16

C \*\*\*\* Compute BETA [= turbine pressure ratio/compressor pressure ratio]

BETA=PR(6)\*PR(7)\*PR(8)\*PR(9)\*PR(10)\*PR(13)\*PR(14)\*PR(15)\*PR(16)\*  
& PR(17)

C \*\*\*\* Turbine Pressure Ratio

PRT=PRC\*BETA

C \*\*\*\* The cycle definition is accomplished by solving for working fluid  
C flowrate and pressure level required to produce the specified  
C power output. Key parameters include:

C N - rotor speed, rpm  
C Ns,c - compressor inlet specific speed, rpm-ft^.75/sec^.5  
C PRESS(1) - compressor inlet pressure, psia  
C FLOW(1) - compressor inlet flowrate, lbm/sec

C N and Ns,c are specified; FLOW(1) and PRESS(1) are determined  
C \*\*\*\* to provide the specified output power level.

TEMP(2)=((PRC\*\*((GAMMA-1)/GAMMA)-1)/EFFC+1)\*TEMP(1)  
PRESS(4)=PRESS(1)\*PRC

IF (COMPTYPE.EQ.'RADIAL') THEN  
IF (COMPSS.LE.45.) THEN  
COMPSS=45.  
VAR(5)=COMPSS  
END IF  
IF (COMPSS.GT.100.) THEN  
COMPSS=100.  
VAR(5)=COMPSS  
END IF  
END IF

IF (COMPTYPE.EQ.'AXIAL') THEN

```

IF (COMPSS.LT.(33-1.4*PRC)) THEN
COMPSS=33.-1.4*PRC
VAR(5)=COMPSS
END IF
IF (COMPSS.GT.(92/(PRC-1)**.675)) THEN
COMPSS=92/(PRC-1)**.675
VAR(5)=COMPSS
END IF
END IF

FLOW(1)=144.*MOLWT/RU*SQRT(PRESS(1)*PRESS(4)/(TEMP(1)*TEMP(2)))*
& (COMPSS/SPEED)**2*(JCON*CP*(TEMP(2)-TEMP(1)))**1.5

```

C \*\*\*\* Flows

```

BLEEDC=XBLC*FLOW(1)
BLEEDT=XBLT*FLOW(1)
FLOW(1)=FLOW(1)-BLEEDC
FLOW(2)=FLOW(1)
FLOW(3)=FLOW(2)+BLEEDC
FLOW(4)=FLOW(3)
FLOW(5)=FLOW(3)-BLEEDC-BLEEDT
FLOW(6)=FLOW(5)
FLOW(7)=FLOW(6)
FLOW(8)=FLOW(7)
FLOW(9)=FLOW(8)
FLOW(10)=FLOW(9)
FLOW(11)=FLOW(10)
FLOW(12)=FLOW(11)
FLOW(13)=FLOW(12)+BLEEDT
FLOW(14)=FLOW(13)
FLOW(15)=FLOW(14)
FLOW(16)=FLOW(15)
FLOW(17)=FLOW(16)

H1=TEMP(1)*CP
IF (COMPTYPE.EQ.'RADIAL      ') THEN
PR2A=(PRC-1.)/2.+1.
PR2=PRC/PR2A
ELSE
PR2A=PRC
PR2=1.
END IF

TEMP(4)=TEMP(2)
H2=CP*TEMP(4)

PRESS(5)=PRESS(4)
PRESS(2)=(PRESS(5)+PRESS(1))/2
PRESS(3)=PRESS(2)
PRESS(6)=PRESS(5)*PR(6)
PRESS(7)=PRESS(6)*PR(7)

```

```
PRESS(8)=PRESS(7)*PR(8)
PRESS(9)=PRESS(8)*PR(9)
PRESS(10)=PRESS(9)*PR(10)
PRESS(12)=PRESS(1)/PR(13)/PR(14)/PR(15)/PR(16)/PR(17)
PRESS(11)=(PRESS(12)+PRESS(10))/2
PRESS(13)=PRESS(12)*PR(13)
PRESS(14)=PRESS(13)*PR(14)
PRESS(15)=PRESS(14)*PR(15)
PRESS(16)=PRESS(15)*PR(16)
PRESS(17)=PRESS(16)*PR(17)
```

C \*\*\*\* Temperatures

```
TEMP(3)=TEMP(2)
TEMP(5)=TEMP(4)
```

C \*\*\*\* All cooling of windage, generator EM losses and bearing losses  
C is accomplished with a secondary [auxilliary] cooling loop.  
C Therefore there is no temperature rise associated with this  
C \*\*\*\* heat removal.

```
TEMP(6)=TEMP(5)
```

C \*\*\*\* Establish some initial guesses at key bycle temperatures

```
T10G=(1.-(1.-1./PRT**((GAMMA-1)/GAMMA))*EFFT)*TEMP(9)
T4G=TEMP(6)+(T10G-TEMP(6))*EFFR
T5G=T4G
T6G=TEMP(9)
T9G=T10G
```

C \*\*\*\* Enthalpies

```
H6A=CP*TEMP(9)
20 H5=CP*T5G
```

C \*\*\*\* Thermal losses are determined from input fraction of gross  
C \*\*\*\* thermal input power.

C \*\*\*\* Gross thermal input power

```
POWER6A=FLOW(9)*(H6A-H5)
```

C \*\*\*\* Loss in heat source inlet flowpath

```
POWER5=-FCTQ8* POWER6A
```

C \*\*\*\* Loss in turbine inlet flowpath

```
POWER6=-FCTQ10*POWER6A
```

C \*\*\*\* Loss in the turbine discharge flowpath

```
POWER10=-FCTQ13* POWER6A
```

```
C **** Thermal losses in kWt
```

```
QLOSS8=POWER5 * 3600./3413.  
QLOSS10=POWER6 * 3600./3413.  
QLOSS13=POWER10 * 3600./3413.
```

```
C **** Temperatures/Enthalpies
```

```
TEMP(10)=TEMP(9)+POWER6/FLOW(10)/CP  
TEMP(11)=TEMP(10)  
TEMP(12)=(1-(1-1/PRT**((GAMMA-1)/GAMMA))*EFFT)*TEMP(11)  
H9=CP*TEMP(12)  
H10=(H9*FLOW(12)+BLEEDT*H2)/(FLOW(12)+BLEEDT)  
TEMP(13)=H10/CP+POWER10/FLOW(13)/CP  
TEMP(7)=TEMP(6)+(TEMP(13)-TEMP(6))*EFFR  
TEMP(14)=TEMP(13)+(TEMP(6)-TEMP(7))*FLOW(6)/FLOW(13)  
TEMP(8)=TEMP(7)+POWER5/FLOW(8)/CP  
TEMP(15)=TEMP(14)
```

```
C **** Check for convergence of computed and estimated temperatures.
```

```
IF ((ABS(TEMP(12)-T9G)/TEMP(12)).LT..00001.AND.(ABS(TEMP(7)  
& -T4G)/TEMP(7)).LT..00001) GO TO 50
```

```
T4G=TEMP(7)  
T5G=TEMP(8)  
T9G=TEMP(12)  
GO TO 20
```

```
50 TEMP(16)=TEMP(1)  
TEMP(17)=TEMP(1)
```

```
C **** Enthalpies
```

```
H2A=TEMP(2)*CP  
H2B=TEMP(3)*CP  
H2C=TEMP(5)*CP  
H3=TEMP(6)*CP  
H4=TEMP(7)*CP  
H6=TEMP(10)*CP  
H7=TEMP(11)*CP  
H11=TEMP(14)*CP  
H11A=TEMP(15)*CP  
H12=TEMP(16)*CP  
H13=TEMP(17)*CP
```

```
C **** Power Calculations
```

```
POWER2A=-FLOW(2)*(H2A-H1)  
POWER4=-FLOW(7)*(H4-H3)
```

```

POWER9=FLOW(12)*(H7-H9)
POWER11=-FLOW(14)*(H11-H10)
POWER12=-FLOW(16)*(H11-H12)
C **** Power and conversion efficiency calculation
GROSSSP=(POWER9+POWER2A)*3600./3413.

C **** Compute net shafe power
60 NETSP=GROSSSP-WINDAGE-BRGLOSS

CALL GENSIZE
EFFA=ETAGEN

ALTLOSS=(1.-EFFA)*NETSP
GROSSEP=NETSP-ALTLOSS

C **** Bearing loss and windage loss returned in kWs
CALL BEARING
CALL WINDLOS

COOLING=WINDAGE+BRGLOSS+ALTLOSS
EFFCYCLE= GROSSEP/POWER6A*3413/3600

C **** Compute the cooling load resulting from generator EM and windage
C **** losses. Compute the required coolant flowrate.

WCLNT=COOLING*3413./3600./(CPCLNT*(TOUTCLNT-TINCLNT))

70 CONTINUE

C **** Call compressor size/efficiency routine.

IF (COMPTYPE.EQ.'RADIAL') CALL RADCOMP

IF (COMPTYPE.EQ.'AXIAL') CALL AXCOMP
EFFC=MAX(ETACOMP,.60)
EFFC=MIN(ETACOMP,.95)

C **** Call turbine size/efficiency routine.

IF (TURBTYPE.EQ.'RADIAL') CALL RADTURB

IF (TURBTYPE.EQ.'AXIAL') CALL AXTURB
EFFT=MAX(ETATURB,.60)
EFFT=MIN(ETATURB,.95)

C **** If DPOP9 [HSA pressure loss fraction] is a dependent variable, then
C **** Subroutine HSA must be called at this point and DPOP9 updated.
C DP9OLD=DPOP9

```

```

C      CALL HSA
C      DPOP9=DP9
C      VAR(10)=DP9

C **** If DPOP16 [HRA pressure loss fraction] is a dependent variable, then
C **** Subroutine HRA must be called at this point and DPOP16 updated.
C      DP16OLD=DPOP16
C      CALL HRA
C      DPOP16=DP16
C      VAR(14)=DP16

C **** Check for convergence of specified power level and aero/themo
C **** calculated power.
ITER=ITER+1
IF (ABS((POWER-GROSSEP)/POWER).LT..0001.AND.ITER.GT.2) RETURN

C **** Reset system pressure level to attain specified power.

C ***** Error trap inserted following telecon with LeRC on 7-9-93 ***
IF (PRESS(1)*POWER/GROSSEP.LE.0) THEN
  ERRORC=' Negative compressor inlet pressure predicted in STATEPT'
  WRITE (*,100) ERRORC
100 FORMAT (1X,A64)
  RETURN
END IF

C **** **** **** **** **** **** **** **** **** **** **** **** **** **** **** ****

PRESS(1)=PRESS(1)*POWER/GROSSEP
GO TO 10

END

SUBROUTINE RECINP
C **** RECINP is checked-out and final on October 6, 1992
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)

CHARACTER*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18)

COMMON/TITLE/ CYCDES,TITLER
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
COMMON/OPTIM/ VAR(18),TVAR,IPRINT
COMMON/RECUP/ DPOPREC,UAML,UAMH,DPMI,DPMH,
& PREXP,NFINL,TFINL,SPACEL,LEQL,ROFINL,KFINL,NFINH,TFINH,SPACEH,
& LEQH,ROFINH,KFINH,TPLATE,ROPLAT,TBRAZE,ROBRAZ,
& SIGMAL,ALPHAL,AFINL,APLATL,HDL,SIGMAH,ALPHAH,AFINH,APLATH,HDH,
& PITCH,WGTREC
COMMON/CONST/PI,RU,GO,JCON,RZERO

```

```
READ (60,90) (TITLER(I),I=1,8)
READ (60,40) PREXP,UAML,DPML,UAMH,DPMH
READ (60,70) XNFINL,XTFINL,XSPACEL,XLEQL,XROFINL,XKFINL,
& XNFINH,XTFINH,XSPACEH,XLEQH,XROFINH,XKFINH,
& XTPLATE,XROPLAT,
& XTBRAZE,XROBRAZ
```

C \*\*\* Convert to engineering units for calculations

```
NFINL=XNFINL*2.54
TFINL=XTFINL/2.54
SPACEL=XSPACEL/2.54
LEQL=XLEQL/2.54
ROFINL=XROFINL*(2.54**3.)*(1./453.6)
KFINL=XKFINL*3600.*0.3048/(1054.4*1.8)
```

```
NFINH=XNFINH*2.54
TFINH=XTFINH/2.54
SPACEH=XSPACEH/2.54
LEQH=XLEQH/2.54
ROFINH=XROFINH*(2.54**3.)*(1./453.6)
KFINH=XKFINH*3600.*0.3048/(1054.4*1.8)
```

```
TPLATE=XTPLATE/2.54
ROPLAT=XROPLAT*(2.54**3.)*(1./453.6)
TBRAZE=XTBRAZE/2.54
ROBRAZ=XROBRAZ*(2.54**3.)*(1./453.6)
```

C \*\* Initial geometry calcs \*\*\*

```
PITCH=SPACEL+SPACEH+2.*TPLATE
CALL RCTFIN (SPACEL,TFINL,NFINL,PITCH,LEQL,
& SIGMAL,ALPHAL,AFLATL,AFLNL,HDL)
CALL RCTFIN (SPACEH,TFINH,NFINH,PITCH,LEQH,
& SIGMAH,ALPHAH,AFLATH,AFLNH,HDH)
```

C \*\*\*\* Free Flow:Frontal Area Ratios  
C Define Surface Area per Unit Volume (sq in/cu in)  
C Fin Area per Unit Volume (sq in/cu in)  
C Plate Area per Unit Volume (sq in/cu in)  
C \*\*\*\* Define Hydraulic Diameters

C \*\* convert to SI units for output \*\*

```
XSIGMAL=SIGMAL
XALPHAL=ALPHAL/2.54
XAFLNL=AFLNL/2.54
XAFLATL=AFLATL/2.54
XHDL=HDL*2.54
```

```
XSIGMAH=SIGMAH
XALPHAH=ALPHAH/2.54
```

```

XAFINH=AFINH/2.54
XAPLATH=APLATH/2.54
XHDH=HDH*2.54

IF (IPRINT.GE.1) THEN
IF (VAR(4).GT.0.0) THEN
WRITE (61,100)
WRITE (61,210) (TITLER(I),I=1,8)
WRITE (61,220) GAS(1),GAS(2),UAML,DPMI,RZERO,GAS(1),GAS(2),
& UAMH,DPMH,RZERO
WRITE (61,230) XNFINL,XTFINH,XSPACEL,XLEQL,XROFINL,XKFINL,
& XNFINH,XTFINH,XSPACEH,XLEQH,XROFINH,XKFINH,
& XTPLATE,XROPLAT,
& XTBAZE,XROBRAZ
WRITE (61,240) XSIGMAL,XALPHAL,XAFINL,XAPLATL,XHDL,XSIGMAH,
& XALPHAH,XAFINH,XAPLATH,XHDH
END IF
END IF

40 FORMAT (6E12.5)
70 FORMAT (2(6E12.5/),2(12X,E12.5,24X,E12.5/))
90 FORMAT (8A10)
100 FORMAT (33X,'INPUT GEOMETRY FOR COUNTERFLOW HEAT EXCHANGER DESIGN
&PROGRAM',//)
210 FORMAT (50X,8A10/)
220 FORMAT (37X,'GAS      GAS',5X,'UA MARGIN   DP MARGIN
&',2X,'COND. FACTOR',/58X,3('(- - -)',5X),/,'L.P. SIDE -----
&----- ',2X,2A10,3(F9.4,3X),24X,/,'H.P. SIDE -----
&----- ',2X,2A10,3(F9.4,3X)//)
230 FORMAT (36X, 'NUMBER      THICKNESS      SPACING      EQ. LENGTH
&DENSITY      CONDUCTIVITY',/36X, ' (PER CM)      (CM)      (CM)
& (CM)      (GM/CC)      (W/M-K)//,
& ' L.P. SIDE FINS ----- ',F9.2,5F12.3/,,
& ' H.P. SIDE FINS ----- ',F9.2,5F12.3/,,
& ' SPLITTER PLATES ----- ',9X,F12.3,24X,F12.3/,,
& ' BRAZE MATERIAL ----- ',9X,F12.3,24X,F12.3//)

240 FORMAT (//37X, 'FLOW AREA'4X, 'SURFACE'7X, 'FIN'8X, 'PLATE'5X, 'HY
&DRAULIC',/39X, 'RATIO'4X, 'AREA/VOLUME'1X, 'AREA/VOLUME'1X, 'ARE
&A/VOLUME'3X, 'DIAMETER',/38X, '(- - -)'3(5X,'(1/CM )'),5X, '(CM
&')//'
& ' L.P. SIDE FINS ----- ',F9.5,3(3X,F9.3),3X,F9.5/,,
& ' H.P. SIDE FINS ----- ',F9.5,3(3X,F9.3),3X,F9.5/)

RETURN
END

```

SUBROUTINE RCTFIN (SPACE,TFIN,NFIN,PITCH,LEQ,SIGMA,ALPHA,APLATE,
& AFIN,HD)  
C \*\*\*\* RCTFIN is checked-out and final on October 6, 1992

```

IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)

C **** This subroutine calculates the heat transfer and aerodynamic
C **** geometric constants for rectangular fin assemblies.

C **** Calculate the flow area:frontal area ratio

    SIGMA=(SPACE-TFIN)*(1-TFIN*NFIN)/PITCH

C **** Calculate the plate area:volume ratio

    APLATE=2.*(1.-NFIN*TFIN)/PITCH

C **** Calculate the fin area:volume ratio

    AFIN=2.*NFIN*(SPACE-TFIN)/PITCH

C **** Calculate the equivalent fin length

    IF (LEQ.EQ.0.) LEQ=SPACE/2.

C **** Calculate heat transfer area:volume ratio

    ALPHA=APLATE+AFIN

C **** Calculate passage hydraulic diameter

    HD=4.*SIGMA/ALPHA

    RETURN
    END

    SUBROUTINE GASPROP
C **** GASPROP is checked-out and final on October 6, 1992

C **** This subroutine provides input data pertaining to aerodynamic,
C     thermodynamic and transport properties of Noble gases or binary
C **** Noble gas mixtures.

    IMPLICIT REAL (A-H)
    IMPLICIT INTEGER (I-J)
    IMPLICIT REAL (K-Z)
    REAL JCON

    CHARACTER*10 GAS(2)
    CHARACTER*64 ERRORT,ERRORC,ERRORT,ERRORG,ERRORF,WARNINT,WARNINC,
    & WARNINGR,WARNNGG

    COMMON/DIAGNOS/ERRORT,ERRORC,ERRORT,ERRORG,ERRORF,WARNINT,
    & WARNINC,WARNINGR,WARNNGG

```

```

DIMENSION CSC1(79),CSC2(79)
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
COMMON/TRPRO/X(79),Y(79),C(79,3),XGAS1,XGAS2,RMOLWT,
+ SIGGAS1,SIGGAS2,WMOL1,WMOL2,LJPGAS1,LJPGAS2
COMMON/CONST/PI,RU,GO,JCON,RZERO

DATA CSC1/.3,.35,.4,.45,.5,.55,.6,.65,.7,.75,.8,.85,.9,.95,1.,1.05
+,1.1,1.15,1.2,1.25,1.3,1.35,1.4,1.45,1.5,1.55,1.6,1.65,1.7,1.75,
+,1.8,1.85,1.9,1.95,2.,2.1,2.2,2.3,2.4,2.5,2.6,2.7,2.8,2.9,3.,3.1,
+,3.2,3.3,3.4,3.5,3.6,3.7,3.8,3.9,4.,4.1,4.2,4.3,4.4,4.5,4.6,4.7,4.8
+,4.9,5.,6.,7.,8.,9.,10.,20.,30.,40.,50.,60.,70.,80.,90.,100./
DATA CSC2/2.785,2.628,2.492,2.368,2.257,2.156,2.065,1.982,1.908,
+,1.841,1.78,1.725,1.675,1.629,1.587,1.549,1.514,1.482,1.452,1.424,
+,1.399,1.375,1.353,1.333,1.314,1.296,1.279,1.264,1.248,1.234,1.221,
+,1.209,1.197,1.186,1.175,1.156,1.138,1.122,1.107,1.093,1.081,1.069,
+,1.058,1.048,1.039,1.030,1.022,1.014,1.007,.9999,.9932,.9870,.9811,
+,.9755,.9700,.9649,.9600,.9553,.9507,.9464,.9422,.9382,.9343,.9305,
+,.9269,.8963,.8727,.8538,.8379,.8242,.7432,.7005,.6718,.6504,.6335,
+,.6194,.6076,.5973,.5882/

```

C \*\*\*\* SIGGAS1 is the collision diameter of gas#1 in Angstroms.  
C \*\*\*\* SIGGAS2 is the collision diameter of gas#2 in Angstroms.

C \*\*\*\* Lennard-Jones parameters expanded to include Neon, Argon, Xenon  
C for use in working fluid mixtures. Data based on:

C R.C. Reid and T.K. Sherwood, "The Properties of Gases and Liquids",  
C \*\*\*\* McGraw Hill Book Company, New York, 1958.

C \*\*\*\* GAMMA [Cp/Cv] is 5/3 for all Noble gases and gas mixtures.

GAMMA=5./3.

C \*\*\*\* Determine the specific heat [@ const P] of the working fluid.

CP=(GAMMA/(GAMMA-1.))\*RU/JCON/MOLWT

SIGXE=4.055  
SIGKR=3.600  
SIGAR=3.418  
SIGNE=2.789  
SIGHE=2.576

RMOLWT=MOLWT

WMOLXE=131.3  
WMOLHE=4.0026  
WMOLNE=20.183  
WMOLAR=39.944  
WMOLKR=83.8

LJPXE=1./229.

LJPKR=1./190.  
LJPAR=1./124.  
LJPNE=1./35.7  
LJPHE=1./10.22

C \*\*\*\* Calculate the mole fractions required to provide specified MW.

```
IDGAS=0
IF (GAS(1).EQ.'HELIUM      ') THEN
WMOL1=WMOLHE
SIGGAS1=SIGH
LJPGAS1=LJPHE
IDGAS=IDGAS+1
END IF
IF (GAS(2).EQ.'HELIUM      ') THEN
WMOL2=WMOLHE
SIGGAS2=SIGH
LJPGAS2=LJPHE
IDGAS=IDGAS+1
END IF
IF (GAS(1).EQ.'NEON        ') THEN
WMOL1=WMOLNE
SIGGAS1=SIGNE
LJPGAS1=LJPNE
IDGAS=IDGAS+1
END IF
IF (GAS(2).EQ.'NEON        ') THEN
WMOL2=WMOLNE
SIGGAS2=SIGNE
LJPGAS2=LJPNE
IDGAS=IDGAS+1
END IF
IF (GAS(1).EQ.'ARGON       ') THEN
WMOL1=WMOLAR
SIGGAS1=SIGAR
LJPGAS1=LJPAR
IDGAS=IDGAS+1
END IF
IF (GAS(2).EQ.'ARGON       ') THEN
WMOL2=WMOLAR
SIGGAS2=SIGAR
LJPGAS2=LJPAR
IDGAS=IDGAS+1
END IF
IF (GAS(1).EQ.'KRYPTON     ') THEN
WMOL1=WMOLKR
SIGGAS1=SIGKR
LJPGAS1=LJPKR
IDGAS=IDGAS+1
END IF
IF (GAS(2).EQ.'KRYPTON     ') THEN
WMOL2=WMOLKR
```

```

SIGGAS2=SIGKR
LJPGAS2=LJPKR
IDGAS=IDGAS+1
END IF
IF (GAS(1).EQ.'XENON      ') THEN
WMOL1=WMOLXE
SIGGAS1=SIGXE
LJPGAS1=LJPXE
IDGAS=IDGAS+1
END IF
IF (GAS(2).EQ.'XENON      ') THEN
WMOL2=WMOLXE
SIGGAS2=SIGXE
LJPGAS2=LJPXE
IDGAS=IDGAS+1
END IF
IF (IDGAS.NE.2) THEN
ERRORRF=
& 'SUBROUTINE GASPROP DID NOT I.D. TWO GASES IN WORKING FLUID'
WRITE (*,100) ERRORRF
END IF

```

```

IF (GAS(1).NE.GAS(2)) THEN
XGAS1=(MOLWT-WMOL2)/(WMOL1-WMOL2)
ELSE
XGAS1=0.5
END IF
XGAS2=1.-XGAS1

```

C \*\*\*\*\* Perform a cubic spline fit of the Lennard/Jones parameter.

```

DO 200 I=1,79
X(I)=CSC1(I)
Y(I)=CSC2(I)
200 CONTINUE
CALL ICSCCU(X,Y,79,C,79,IER)
RETURN
100 FORMAT (1X,A64)
END

```

FUNCTION VISCOS (T)

C \*\*\*\*\* VISCOS is checked-out and final on October 6, 1992

C \*\*\*\*\* This function evaluates the viscosity of a fluid based on  
C \*\*\*\*\* temperature and gas constituents.

```

IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
REAL MU(2)

```

```

C **** LJPGAS1 is the Lennard/Jones parameter for gas#1.
C **** SIGGAS1 is the collision diameter for gas#1 in Angstroms.

COMMON/TRPRO/X(79),Y(79),CC(79,3),XGAS1,XGAS2,MOLWT,SIGGAS1,
+SIGGAS2,WMOL1,WMOL2,LJPGAS1,LJPGAS2
DIMENSION XX(2),WM(2)

OHMGAS1=0.
OHMGAS2=0.

C **** T is converted to Kelvin; cubic spline is evaluated for two gas
C **** constituents.

ARGGAS1=LJPGAS1*T/1.8
IF(ARGGAS1.LE.100.AND.ARGGAS1.GE.0.3) THEN
CALL ICSEVU(X,Y,79,CC,79,ARGGAS1,OHMGAS1,1,IER)
ELSE
IF(ARGGAS1.GT.100)OHMGAS1=.5882-(ARGGAS1-100.)*.00091
IF(ARGGAS1.LT..3)OHMGAS1=2.785 +(.3-ARGGAS1)*3.14
END IF

ARGGAS2=LJPGAS2*T/1.8
IF(ARGGAS2.LE.100.AND.ARGGAS2.GE.0.3) THEN
CALL ICSEVU(X,Y,79,CC,79,ARGGAS2,OHMGAS2,1,IER)
ELSE
IF(ARGGAS2.GT.100)OHMGAS2=.5882-(ARGGAS2-100.)*.00091
IF(ARGGAS2.LT..3)OHMGAS2=2.785 +(.3-ARGGAS2)*3.14
END IF

VISCL=SQRT(WMOL1*T/1.8)/(SIGGAS1**2*OHMGAS1)*.000026693
VISCL=SQRT(WMOL2*T/1.8)/(SIGGAS2**2*OHMGAS2)*.000026693
XX(1)=XGAS1
XX(2)=XGAS2
MU(1)=VISCL
MU(2)=VISCL
WM(1)=WMOL1
WM(2)=WMOL2
VISCOS=0.
DO 100 I=1,2
RNUMER=XX(I)*MU(I)
DENOM2=0.
DO 200 J=1,2
DENOM=XX(J)*(1.+(MU(I)/MU(J))**.5*(WM(J)/WM(I))**.25)**2
+/8.*.5/((1.+WM(I)/WM(J))**.5)
DENOM2=DENOM2+DENOM
200 CONTINUE
VISCOS=VISCOS+RNUMER/DENOM2
100 CONTINUE

C **** Convert the viscosity from g/cm-sec to lbm/ft-sec.

VISCOS=VISCOS*.0672

```

```
30 CONTINUE
RETURN
END
```

```
FUNCTION AVVISC (TONE,TTWO)
```

```
C **** AVVISC is checked-out and final on October 6, 1992
C **** This function evaluates the average viscosity of a gas or gas
C **** mixture over the temperature range from TONE to TTWO using Simpson's
C **** Rule.
```

```
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
```

```
TAVG=(TONE+TTWO)/2.
```

```
AVVISC =(VISCOS(TONE)+4.*VISCOS(TAVG)+VISCOS(TTWO))/6.
```

```
RETURN
END
```

```
FUNCTION CONDUCT (T)
```

```
C **** RECSIZE is checked-out and final on October 6, 1992
```

```
C **** This function evaluates the thermal conductivity of a gas or
C **** gas mixture based on the temperature (T) and gas constituents.
```

```
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
```

```
C **** LJPGAS1 is the Lennard/Jones parameter for gas#1
C **** SIGGAS1 is the collision diameter for gas#1 in Angstoms.
```

```
COMMON/TRPRO/X(79),Y(79),CC(79,3),XGAS1,XGAS2,MOLWT,
+SIGGAS1,SIGGAS2,WMOL1,WMOL2,LJPGAS1,LJPGAS2
```

```
DIMENSION CON(2),XX(2),VISC(2),WM(2)
```

```
OHMGAS1=0.
```

```
OHMGAS2=0.
```

```
ARGGAS1=LJPGAS1*T/1.8
```

```
IF(ARGGAS1.GT.100.OR.ARGGAS1.LT..3)GO TO 21
```

```
C **** T is converted from R to K; the cubic spline is evaluated/
C **** If the argument is outside the valid range of the fit, then
C **** use linear extrapolation.
```

```
CALL ICSEVU(X,Y,79,CC,79,ARGGAS1,OHMGAS1,1,IER)
```

```
21 IF(ARGGAS1.GT.100)OHMGAS1=.5882-(ARGGAS1-100.)*.00091
```

```
IF(ARGGAS1.LT..3)OHMGAS1=2.785 +(.3-ARGGAS1)*3.14
```

```
ARGGAS2=LJPGAS2*T/1.8
```

```
IF(ARGGAS2.GT.100.OR.ARGGAS1.LT..3)GO TO 25
```

```

CALL ICSEVU(X,Y,79,CC,79,ARGGAS2,OHMGAS2,1,IER)
25 IF(ARGGAS2.GT.100)OHMGAS2=.5882-(ARGGAS2-100.)*.00091
IF(ARGGAS2.LT..3)OHMGAS2=2.785 +(.3-ARGGAS2)*3.14
COND1=(T/WMOL1/1.8)**.5/(SIGGAS1**2*OHMGAS1)*.00019891
COND2=(T/WMOL2/1.8)**.5/(SIGGAS2**2*OHMGAS2)*.00019891
VISCI=(WMOL1*T/1.8)**.5/(SIGGAS1**2*OHMGAS1)*.000026693
VISC2=(WMOL2*T/1.8)**.5/(SIGGAS2**2*OHMGAS2)*.000026693
XX(1)=XGAS1
XX(2)=XGAS2
VISC(1)=VISCI
VISC(2)=VISC2
CON(1)=COND1
CON(2)=COND2
WM(1)=WMOL1
WM(2)=WMOL2
CONDUCT=0.
DO 100 I=1,2
RNUMER=XX(I)*CON(I)
DENOM2=0.
DO 200 J=1,2
DENOM=XX(J)*(1.+(VISC(I)/VISC(J))**.5*(WM(J)/WM(I))**.25)**2
+/8.*.5/((1.+WM(I)/WM(J))**.5)
DENOM2=DENOM2+DENOM
200 CONTINUE
CONDUCT=CONDUCT+RNUMER/DENOM2
100 CONTINUE

```

C \*\*\*\* Convert the conductivity from cal/cm-sec -K to BTU/ft-sec-R

CONDUCT=CONDUCT\*.0671985

RETURN  
END

FUNCTION AVCOND (TONE,TTWO)

C \*\*\*\* AVCOND is checked-out and final on October 6, 1992  
C \*\*\*\* This function evaluates the average thermal conductivity of a gas  
C \*\*\*\* over the temperature range from TONE to TTWO using Simpson's Rule.

IMPLICIT REAL (A-H)  
IMPLICIT INTEGER (I-J)  
IMPLICIT REAL (K-Z)

TAVG=(TONE+TTWO)/2.  
AVCOND=(CONDUCT(TONE)+4.\*CONDUCT(TAVG)+CONDUCT(TTWO))/6.

RETURN  
END

```

SUBROUTINE ICSCCU (X,Y,NX,C,IC,IER)
DIMENSION X(NX),Y(NX),C(IC,3)

C **** Implementation of cubic spline with interface of IMSL routine
C      using SPLINE routine from "Computer methods for Mathematical
C **** Computations" by Forsythe, Malcolm, and Moler.

IER=0
IF (IC.LT.NX) THEN
IER=129
RETURN
END IF
IF (NX.LT.2) THEN
IER=130
RETURN
END IF
DO 10 I=1,NX-1
IF (X(I).GE.X(I+1)) THEN
IER=131
RETURN
END IF
10 CONTINUE
CALL SPLINE (NX,X,Y,C(1,1),C(1,2),C(1,3))
RETURN
END

SUBROUTINE ICSEVU(X,Y,NX,C,IC,U,S,M,IER)
DIMENSION X(NX),Y(NX),C(IC,3),U(M),S(M)

C **** Implementation of cubic spline with interface of IMSL routine
C      using SEVAL routine from "Computer methods for Mathematical
C **** Computations" by Forsythe, Malcolm, and Moler.

IER=0
DO 10 I=1,M
IF (U(I).LT.X(1)) THEN IER=33
IF (U(I).GT.X(NX)) THEN IER=34
S(I)=SEVAL(NX,U(I),X,Y,C(1,1),C(1,2),C(1,3))
10 CONTINUE
RETURN
END

SUBROUTINE SPLINE (N,X,Y,B,C,D)
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
INTEGER N
REAL X(N),Y(N),B(N),C(N),D(N)
INTEGER I
REAL T

IF (N.LT.2) RETURN

```

```

IF (N.LT.3) GOTO 50

C ***** Set up tri-diagonal system
C ***** B = DIAGONAL, D = OFFDIAGONAL, C = RIGHT HAND SIDE.

D(1)=X(2)-X(1)
C(2)=(Y(2)-Y(1))/D(1)
DO 10 I=2,N-1
D(I)=X(I+1)-X(I)
B(I)=2.*(D(I-1)+D(I))
C(I+1)=(Y(I+1)-Y(I))/D(I)
C(I)=C(I+1)-C(I)

10 CONTINUE

C ***** End conditions. Third derivatives at X(1) and X(N)
C ***** are obtained from divided differences.

B(1)=-D(1)
B(N)=-D(N-1)
C(1)=0.0
C(N)=0.0
IF (N.EQ.3) GO TO 15
C(1)=C(3)/(X(4)-X(2))-C(2)/(X(3)-X(1))
C(N)=C(N-1)/(X(N)-X(N-2))-C(N-2)/(X(N-1)-X(N-3))
C(1)=C(1)*D(1)**2/(X(4)-X(1))
C(N)=-C(N)*D(N-1)**2/(X(N)-X(N-3))

C ***** Forward elimination

15 DO 20 I=2,N
    T=D(I-1)/B(I-1)
    B(I)=B(I)-T*D(I-1)
    C(I)=C(I)-T*C(I-1)
20 CONTINUE

C ***** Back substitution

C(N)=C(N)/B(N)
DO 30 I=N-1,1,-1
    C(I)=(C(I)-D(I)*C(I+1))/B(I)
30 CONTINUE

C ***** Compute the polynomial coefficients

B(N)=(Y(N)-Y(N-1))/D(N-1)+D(N-1)*(C(N-1)+2.*C(N))
DO 40 I=1,N-1
    B(I)=(Y(I+1)-Y(I))/D(I)-D(I)*(C(I+1)+2.*C(I))
    D(I)=(C(I+1)-C(I))/D(I)
    C(I)=3.*C(I)
40 CONTINUE
    C(N)=3.*C(N)

```

```

D(N)=D(N-1)
RETURN

50 B(1)= (Y(2)-Y(1))/(X(2)-X(1))
C(1)=0.0
D(1)=0.0
B(2)=B(1)
C(2)=0.0
D(2)=0.0
RETURN
END

FUNCTION SEVAL(N, U, X, Y, B, C, D)

IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
INTEGER N
REAL U,X(N),Y(N),B(N),C(N),D(N)
INTEGER I,J,K
REAL DX
      d
DATA I/1/

IF (I.GE.N) I=1
IF (U.LT.X(I)) GO TO 10
IF (U.LE.X(I+1)) GO TO 30

C ***** Conduct binary search.

10 I=1
J=N+1
20 K=(I+J)/2
IF (U.LT.X(K)) J=K
IF (U.GE.X(K)) I=K
IF (J.GT.I+1) GO TO 20

C ***** Evaluate the spline function.

30 DX=U-X(I)
SEVAL=Y(I)+DX*(B(I)+DX*(C(I)+DX*D(I)))
RETURN
END

C INTERMEDIATE HEAT EXCHANGER SIZING CODE
C
SUBROUTINE IHXSHL(IHXflg,UEST,THIN,THOUT,PHOT,TCIN,TCOUT,WDOTS,
&AMWS,TINS,DENINS,DENSSH,DTUBE,PR,TTUBE,ANPLATES,WDOTT,AKTUBE,Q
&DOT,DPSHELL,ANTUBES,DPTUBE,DTL2,ALSHEL,AMSHELL,AMPLATES,
&AMTUBES,AMINSUL,AMHEADS,AMSTRT,ANETMASS,XMNHEX,HSHELL,AFRIC,UNEW,
&RETUBE,THC,AMTSHT)

```

```

C      WRITE (6,*)
C      WRITE (6,*) 'DATA INPUT LIST FROM IHXSHL'
C      WRITE (6,*) 'IHXflg,UEST,THIN,THOUT,PHOT,TCIN,TCOUT,WDOTS,AMWS,
C      &TINS,DENINS,DENSSH,DTUBE,PR,TTUBE,ANPLATES,WDOTT,AKTUBE,QDOT =',
C      &IHXflg,UEST,THIN,THOUT,PHOT,TCIN,TCOUT,WDOTS,AMWS,TINS,DENINS,
C      &DENSSH,DTUBE,PR,TTUBE,ANPLATES,WDOTT,AKTUBE,QDOT
C
C      SHELL AND TUBE HEAT EXCHANGER DESIGN SUBROUTINE
C      ROUTINE ASSUMES THAT LIQUID IS ON TUBE SIDE
C      GAS IS ON SHELL SIDE.  BELL'S CORRELATION IS USED FOR
C      GAS SIDE HEAT TRANSFER - LYONS IS USED FOR TUBE SIDE
C      LIQUID METAL HEAT TRANSFER, McELLIOT,MCGEE AND LEPPERT
C      IS USED FOR OTHER FLUIDS (LIQUIDS AND GASES)
C
C      ***** OVERALL PARAMETERS *****
C
C      IHXflf = 1, THEN TUBE SIDE FLUID IS LITHIUM
C      IHXflg = 2, THEN TUBE SIDE FLUID IS NaK-78
C      ALMTD = Heat Exchanger Log Mean Temperature Difference
C      QDOT = Heat Rate or Duty (BTU/Hr)
C      UEST = INITIAL VALUE OF Overall (BTU/Hr-Ft-R)
C          (50 for GAS-GAS)
C      THIN = HOT SIDE Inlet Temperature (R)
C      THOUT = HOT SIDE Outlet Temperature (R)
C      TCIN = COLD SIDE Inlet Temperature (R)
C      COUT = COLD SIDE Outlet Temperature (R)
C
C      ***** SHELL SIDE DATA *****
C
C      WDOTS = SHELL SIDE FLUID Flowrate (Lbs/Sec)
C      DENSSH = SHELL MATERIAL Density (Lbs/Ft^3)
C
C      ***** SHELL-SIDE FLUID PROPERTIES *****
C
C      CPSF = SHELL-SIDE FLUID Specific Heat (BTU/Lb-R)
C      RHOSF = SHELL-SIDE FLUID Density (Lbs/Ft^3)
C      AKTST = SHELL-SIDE FLUID Thermal Cond (BTU/Hr-Ft-R)
C      VISCST = SHELL-SIDE FLUID Viscosity (Cp)
C
C      ***** TUBE SIDE DATA *****
C
C      DTUBE = Outside TUBE Diameter - (Inches)
C      TTUBE = TUBE Wall Thickness (Inches)
C      WDOTT = TUBE -SIDE Fluid Flowrate (Lbs/Sec)
C      AKTUBE = TUBE Wall Thermal Conductivity (BTU/Hr-Ft-R)
C
C      ***** TUBE-SIDE FLUID PROPERTYIES
C
C      CPT = TUBE-SIDE FLUID Specific Heat (BTU/Lb-R)
C      RHOT = TUBE-SIDE FLUID Density (Lbs/Ft^3)
C      AKTT = TUBE-SIDE FLUID Thermal Cond (BTU/Hr-Ft-R)
C      VISCT = TUBE-SIDE FLUID Viscosity (Lb/Ft-Sec)

```

```

C
C      SET FLUID THERMAL PROPERTIES
C
C      TBARR = (THIN+TCIN)/2.0
C      GO TO (5,8),IHXflg
5   CALL XLITHP(TBARR,RHOT,CPT,VISCT,AKTT)
     CALL HEXEPR(AMWS,PHOT,TBARR,GAMMA,CPSF,RHOSF,VISCST,AKTST,PRMIX)
     GO TO 10
8   CALL XNAKPR(TBARR,RHOT,CPT,VISCT,AKTT)
     CALL HEXEPR(AMWS,PHOT,TBARR,GAMMA,CPSF,RHOSF,VISCST,AKTST,PRMIX)
10  WDOTS = 3600.0*WDOTS
     VISCST = VISCST/6.72E-4
     A1 = (THIN-TCOUT) - (THOUT-TCIN)
     IF ((THIN-TCOUT).EQ.(THOUT-TCIN)) THEN
       ALMTD = THIN-TCOUT
     GO TO 15
     ELSE
       A2 = ALOG((THIN-TCOUT)/(THOUT-TCIN))
       ALMTD = A1/A2
     ENDIF
15  PR = PR
     WRITE (6,*) 'A1, ALMTD, PR =', A1, ALMTD, PR
     GOTO 100
35  UEST = UNEW
100 AQ = -2.0*DTUBE
     BQ = (DTUBE**2.0)
     CQ1 = PR*DTUBE
     CQ2 = 144.0*QDOT*(CQ1**2.0)
     CQ3 = UEST*ALMTD*(9.869604)*DTUBE
     CQ = CQ2/CQ3
     PQ = -((AQ**2.0)/3.0)+BQ
     QQ = (2.0*((AQ/3.0)**3.0))-(AQ*BQ/3.0)+CQ
     QBIG = ((PQ/3.0)**3.0)+((QQ/2.0)**2.0)
     AARG = -(QQ/2.0)+SQRT(QBIG)
     ABIG = (ABS(AARG))**0.333333
     BBRG = -(QQ/2.0)-SQRT(QBIG)
     BBIG = (ABS(BBRG))**0.333333
     DOTL = ABIG + BBIG - (AQ/3.0)
     ALSHEL = (ANPLATES+1.0)*DOTL
     AVAL = 0.867
     FFBN = 0.4307652 + (4.521962E-03*(DOTL)) - (6.335725E-05*(DOTL**2
1.0)) + (3.716571E-07*(DOTL**3.0))
     GXT = 1560.0*(WDOTS/(DOTL**2.0))
     REXT = (DTUBE*GXT)/(29.0*VISCST)
     RC = 1.0
     IF (REXT-1000.0) 200,150,150
150 RC = 1.0
     GOTO 300
200 RC = 0.4812508 + (2.726048E-03*(REXT)) - (7.739889E-06*(REXT**2.))
     1) + (9.960931E-09*(REXT**3.0)) - (4.431738e-12*(REXT**4.0))
300 FFBP = FFBN*RC

```

```

FFBH = FFBP+0.125
REXP = FFBP*REXT
REXH = FFBH*REXT
AREP = ALOG(REXP)/2.302585093
AREH = ALOG(REXH)/2.302585093
AFX = 1.397542 - (0.96108*(AREP)) + (0.064751*(AREP**2.0)) + (0.0
106305*(AREP**3.0))
AFRIC = 10.0**AFX
AJX = -0.359018 - (0.259608*(AREH)) - (0.094385*(AREH**2.0)) + (0
1.012556*(AREH**3.0))
AJFAC = 10.0**AJX
HS1 = 0.415*CPSF*GXT*FFBH*AJFAC
HS2 = (AKTST/(CPSF*VISCT))**0.66667
HSHELL = HS1*HS2
DPSF = 0.00875*((AJFAC*ALSHEL)/(AVAL*PR*DTUBE*0.4))
DPSM = 0.001551191*((ALSHEL*1.0)/(0.4*DOTL))-1.0
DPSHELL = (0.3*(DPSF+DPSM)/RHOSF)*((GXT*FFBP)/10000.0)
PT = PR*DTUBE
ANTUBES = ((0.7854*(DOTL-DTUBE)**2.0)/(PT**2.0))
WTUBE = WDOTT/ANTUBES
AREAT = (0.7854*(DTUBE-(2.0*TTUBE))**2.0)/144.0
VTUBE = WTUBE/(AREAT*RHOT)
QTUBE = RHOT*(VTUBE**2.0)/(2.0*32.174*144.0)
RETUBE = VTUBE*RHOT*DTUBE/(12.0*VISCT)
DTUBEI = DTUBE-(2.0*TTUBE)
PRTUBE = (3600*VISCT*CPT)/AKTT
IF (PRTUBE-0.1) 330,330,340
330 THC = (12.0*AKTT/DTUBEI)*(7.0+(0.025*(RETUBE*PRTUBE)**0.8))
GOTO 440
340 IF (RETUBE-2000.0) 350,350,400
350 TFRIC=64.0/RETUBE
    THC = 4.364*(12.0*AKTT/DTUBEI)
    GOTO 450
400 AK = 0.0001
    AKD = AK/DTUBEI
    FRIC1 = ALOG10((AKD/3.7)+(5.74/(RETUBE**0.9)))**2.0
    TFRIC = 0.25/FRIC1
    THC = 0.026*(12.0*AKTT/DTUBEI)*(RETUBE**0.8)*(PRTUBE**0.4)
435 GOTO 450
440 IF (RETUBE-2000.0) 442,442,445
442 TFRIC=64.0/RETUBE
    GOTO 450
445 AK = 0.0001
    AKD = AK/DTUBEI
    FRIC1 = ALOG10((AKD/3.7)+(5.74/(RETUBE**0.9)))**2.0
    TFRIC = 0.25/FRIC1
450 COND2 = 12.0*AKTUBE/TTUBE
    UNEW = 1.0/((1.0/THC)+(1.0/HSHELL)+(1.0/COND2))
    DPTUBE = (2.0*QTUBE) + ((TFRIC*(ALSHEL/DTUBEI))*QTUBE)
    CQ3 = UNEW*ALMTD*(9.869604)*DTUBE
    CQ = CQ2/CQ3
    QQ = (2.0*((AQ/3.0)**3.0))-(AQ*BQ/3.0) + CQ

```

```

QBIG = ((PQ/3.0)**3.0) + ((QQ/2.0)**2.0)
AARG = -(QQ/2.0)+SQRT(QBIG)
ABIG = (ABS(AARG))**0.333333
BBRG = -(QQ/2.0)-SQRT(QBIG)
BBIG = (ABS(BBRG))**0.333333
DOTL2 = ABIG + BBIG - (AQ/3.0)
ERROR = ABS((DOTL-DOTL2)/DOTL)
IF (ERROR - 0.001) 600,600,35
600 TMINPR = PHOT*DOTL/(2.0*10000.0)
TMINSC = (0.005*DOTL) + (0.0001*(DOTL**2.0))
IF (TMINPR.GT.TMINSC) THEN
  TMIN = TMINPR
  GOTO 650
ELSE
  TMIN = TMINSC
ENDIF
650 AMINSUL = (3.141593*DOTL*(DOTL+ALSHEL)*TINS*(DENINS/1728.0)) + (3.
&141593*(DOTL2**2.0)*TINS*(DENINS/1728.0))
AMHEADS = 3.141593*(DOTL**2.0)*TMIN*(DENSSH/1728.0)
AMSHELL = 3.141593*DOTL*(DOTL+ALSHEL)*TMIN*(DENSSH/1728.0)
AMPLATES = ANPLATES*(2.0*3.141593*((DOTL**2.0)/4.0)*TMIN*(DENSSH/1
&1728.0))
AMTSHT = 2.0*3.141593*((DOTL**2.0)/4.0)*2.0*TMIN*(DENSSH/1728.0)
AMTUBES = 0.785398*((DTUBE**2.0)-(DTUBEI**2.0))*ALSHEL*ANTUBES*(1
DENSSH/1728.0)
AMSTRT = 0.05*(AMINSUL+AMHEADS+AMSHELL+AMPLATES+AMTSHT+AMTUBES)
ANETMASS = AMINSUL+AMHEADS+AMSHELL+AMPLATES+AMTSHT+AMTUBES+AMSTRT
VNAK1 = 0.785398*(DTUBE**2.0)*ALSHEL*ANTUBES
VNAK2 = 0.523599*(DOTL**3.0)
XMNHEX = (VNAK1+VNAK2)*(RHOT/1728.0)
RETURN
END

```

#### SUBROUTINE XLITHP(T,RHO,CP,VIS,TK)

```

C THERMAL PROPERTIES OF LITHIUM LIQUID
C T = INPUT TEMPERATURE (deg-R)
C RHO = DENSITY (Lbs/cu-Ft)
C CP = SPECIFIC HEAT (BTU/LB-R)
C VIS = DYNAMIC VISCOSITY (Lb/Ft-Sec)
C TK = THERMAL CONDUCTIVITY (BTU/Hr-Ft-Sec)
C
RHO = 34.393537 - (0.003456*T) + (2.080291E-07*(T**2.0))
CP = 1.356357 - (0.00068*T) + (5.006625E-07*(T**2.0)) - (1.805873E
&-10*(T**3.0)) + (3.155294E-14*(T**4.0)) - (2.136471E-18*(T**5.0))
VIS = 0.001085 - (1.326497E-06*T) + (7.245662E-10*(T**2.0)) - (1.7
&74380E-13*(T**3.0)) + (1.632610E-17*(T**4.0))
TK = 25.235376 + (0.001588*T)
RETURN
END
C

```

```

C
C
C      SUBROUTINE XNAKPR(T,RHO,CP,VIS,TK)
C      THERMAL PROPERTIES OF NaK LIQUID
C      T = INPUT TEMPERATURE (deg-R)
C      RHO = DENSITY (Lb/cu-Ft)
C      CP = SPECIFIC HEAT (BTU/Lb-R)
C      VIS = DYNAMIC VISCOSITY (Lb/Ft-Sec)
C      TK = THERMAL CONDUCTIVITY (BTU/Hr-Ft-R)
C      RHO = 58.54299-(0.008208*T)
C      CP = 0.26478 - (0.000089*(T)) + (4.093060E-08*(T**2.0)) -
C      & (4.532164E-12*(T**3.0))
C      VIS = 0.000822 - (1.142435E-06*(T)) + (6.125737E-10*(T**2.0)) -
C      & (1.130181E-13*(T**3.0))
C      TK = 7.313351 + (0.013983*(T)) - (7.660423E-06*(T**2.0)) +
C      & (1.189370E-09*(T**3.0))
C      RETURN
C      END
C
C
C      SUBROUTINE HEXEPR(Amwmix,Pmix,Tmix,Gma,Cpmix,Rhomix,Amumix,Akmix,
&Prmix)
C      PROPERTIES OF HELIUM-XENON MIXTURES
C      T IN deg-R
C      WRITE (6,*) 'Amwmix,Pmix,Tmix=',Amwmix,Pmix,Tmix
C      Gma=1.667
C      Am1=4.0
C      Am2=131.3
C      X2=(Amwmix-Am1)/(Am2-Am1)
C      X1=1.0-X2
C      Cpmix=4.97/Amwmix
C      Rhomix=144.0*Pmix*Amwmix/(1545.0*Tmix)
C      IF (Tmix.GT.1000.0) THEN
C      GOTO 10
C      ELSE
C      Amuhe=5.7E-06+1.45E-08*Tmix
C      GOTO 100
C      ENDIF
10   IF (Tmix.GT.1600.0) THEN
C      GOTO 20
C      ELSE
C      Amuhe=8.867E-06+1.1333E-08*Tmix
C      GOTO 100
C      ENDIF
20   Amuhe=1.1114E-05+9.930E-09*Tmix
100  IF (Tmix.GT.1100.0) THEN
C      GOTO 30
C      ELSE
C      Akhe=0.0403+8.471E-05*Tmix
C      GOTO 200
C      ENDIF

```

```

30 IF (Tmix.GT.1800.0) THEN
  GOTO 40
ELSE
  Akhe=0.0589+6.786E-05*Tmix
  GOTO 200
ENDIF
40 Akhe=0.0625+6.583E-05*Tmix
200 IF (Tmix.GT.1200.0) THEN
  GOTO 50
ELSE
  Amuxe=5.25E-06+2.1375E-08*Tmix
  GOTO 300
ENDIF
50 IF (Tmix.GT.2000.0) THEN
  GOTO 60
ELSE
  Amuxe=1.0500E-05+1.7000E-08*Tmix
  GOTO 300
ENDIF
60 Amuxe=1.5500E-05+1.45E-08*Tmix
300 IF (Tmix.GT.1200.0) THEN
  GOTO 70
ELSE
  Akxe=0.00115+4.375E-06*Tmix
  GOTO 400
ENDIF
70 IF (Tmix.GT.2500.0) THEN
  GOTO 80
ELSE
  Akxe=0.00252+3.2308E-06*Tmix
  GOTO 400
ENDIF
80 Akxe=0.00342+1.8000E-06*Tmix
400 Amu1=Amuhe
  Amu2=Amuxe
  Ak1=Akhe
  Ak2=Akxe
  Dum1=2.82843*SQRT(1.0+(Am1/Am2))
  Dum2=1.0+SQRT(Amu1/Amu2)*(Am2/Am1)**0.25
  Psi12=Dum2**2.0/Dum1
  Dum3=Amu2/Amu1*(Am1/Am2)
  Psi21=Dum3*Psi12
  Dum4=1.0+Psi12*(X2/X1)
  Dum5=1.0+Psi21*(X1/X2)
  Amumix=Amu1/Dum4+Amu2/Dum5
  Dum6=2.82843*SQRT(1.0+(Am1/Am2))
  Dum7=(1.0+SQRT(Ak1/Ak2)*(Am1/Am2)**0.25)**2.0
  Dum8=2.82843*SQRT(1.0+(Am2/Am1))
  Dum9=(1.0+SQRT(Ak2/Ak1)*(Am2/Am1)**0.25)**2.0
  Alam12=Dum7/Dum6
  Alam21=Dum9/Dum8
  Dum10=(Am1+Am2)**2.0

```

```
Dum11=2.41*(Am1-Am2)*(Am1-0.142*Am2)
Dum12=2.41*(Am2-Am1)*(Am2-0.142*Am1)
For12=Alam12*(1.0+Dum11/Dum10)
For21=Alam21*(1.0+Dum12/Dum10)
Dum13=1.0+For12*(X2/X1)
Dum14=1.0+For21*(X1/X2)
Akmix=Ak1/Dum13+Ak2/Dum14
Prmix=3600*Amumix*Cpmix/Akmix
END
```

**BRAY2.FOR**

```

$DEBUG
$NOTRUNCATE
      SUBROUTINE WINDLOS
C **** WINDLOS is checked-out and final as of 10-06-92

IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
REAL JCON

CHARACTER*10 GAS(2),COMPTYPE,TURBTYPE,CLNTTYPE

CHARACTER*20 GENTYPE,INTTYPE
CHARACTER*64 ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,WARNIGC,
& WARNIGR,WARNIGG

COMMON/DIAGNOS/ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,
& WARNIGC,WARNIGR,WARNIGG

COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDL
COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLOW,
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,
& DPOP16,DPOP17,NETSP,GROSSE,EFFCYCLE,WINDAGE
COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,
& CPCLNT,LIFETIME
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
& RADNC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,
& RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT
COMMON/CONST/PI,RU,GO,JCON,RZERO

TW=960.
PW=PRESS(1)
RHOW=PW*144./(RU/MOLWT*TW)
OMEGA=SPEED*PI/30.
GAPW=DGENRTR*.016
REW=RHOW*GAPW/12.*OMEGA*(DGENRTR/2.)/12./VISCOS(TW)

WCONST=.0406
EXPNT=-.2937

LAMDAW=WCONST*REW**EXPNT
WINDAGE=LAMDAW*RHOW*PI*OMEGA**2*((DGENRTR/2.)/12.)**4
& *(LGENTOT/12.)/GO/JCON*OMEGA* 3600./3413.

RETURN
END

```

SUBROUTINE BEARING

C \*\*\*\* BEARING is checked-out and final as of 10-06-92

\$DEBUG

IMPLICIT REAL (A-H)  
IMPLICIT INTEGER (I-J)  
IMPLICIT REAL (K-Z)

CHARACTER\*10 COMPTYPE,TURBTYPE,CLNTTYPE

CHARACTER\*20 GENTYPE,INTTYPE

COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE

COMMON/GASBRG/BRGLOSS

COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLOW,  
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,  
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,  
& DPOP16,DPOP17,NETSP,GROSSE,EFFCYCLE,WINDAGE

BRGLOSS=0.02\*POWER

RETURN

END

SUBROUTINE GENSIZE

C \*\*\*\* GENSIZE is checked-out and final as of 10-06-92

\$DEBUG

C \*\*\*\* This subroutine utilizes performance, sizing, and mass algorithms  
C to formulate geometry, mass and proportion of a Ring-Wound two-pole  
C toothless [TPTL] PM generator. The algorithms are based on detailed  
C point designs generated by AiResearch Los Angeles Division [ALAD]  
C during May-June of 1992. The coefficients used in this subroutine  
C are derived from the reference designs for implementation into  
C classical "first-principle"  $N*D^{**2}L$  scaling laws for generators.  
C The routine may generate warning messages in the variable "WARNINGG".  
C Warnings are used where some concern exists over the accuracy of  
C calculated values due to the operating domain. Error messages are  
C also produced in variable "ERRORG". Error messages generally  
C indicate input data errors or grossly out-of-range parameters.  
C Calculation is halted and the "RETURN" command executed when an  
C error message is generated. The overall efficiency predicted  
C includes consideration of a small windage loss associated with  
C Potassium vapor in the gap at a temperature of 500 F. The technology  
C assumed is based on commercial materials available today.

C \*\*\*\*\* Nomenclature

C CLNTTYPE	CH*10	Generator Coolant Descripptr, e.g., 'N-HEPTANE'
C COE	REAL	Current Rotor Sizing Coefficient [=f(TIPSPDG,KVA)]
C COMPTYPE	CH*10	Compressor Type, either 'AXIAL' or 'RADIAL'
C COOLING	REAL	Generator Induced Cooling Load [EM+windage+beargs], kWt
C CPCLNT	REAL	Coolant Specific Heat, BTU/(lbm-R)

C	DGENRTR	REAL	Generator Rotor Diameter Including Sleeve, inches
C	DGENSTR	REAL	Generator Stator Diameter Including Sleeve, inches
C	ERRORG	CH*64	Generator Subroutine Diagnostic Error Message
C	GENASP	REAL	Generator Rotor Aspect Ratio, Sleeve OD/Magnet Length
C	GENTYPE	CH*20	Generator Type Descriptor 'RING WOUND TPTL PMG'
C	INTTYPE	CH*20	Electrical Interface Descriptor, either 'TRANSFORMER' or 'RECTIFIER'
C	KVA	REAL	Generator Output, kVA
C	LGENTOT	REAL	Overall Length of Generator EM Section, inches
C	LIFETIME	REAL	Generator Design Lifetime, years
C	MASSEM	REAL	Estimated Generator Electro-magnetic Mass, 1bm
C	MASSGEN	REAL	Estimated Generator Mass w/ Integration Allowance, 1bm
C	OLDTS	REAL	Old Tip Speed Value [used for convergence test], ft/sec
C	PI	REAL	3.14159265
C	POWER	REAL	Generator Terminal Power, kW
C	PWRFCTR	REAL	Load Power Factor at Generator Terminals [lagging]
C	SPEED	REAL	Generator Design Rotating Speed, rpm
C	TINCLNT	REAL	Specified Generator Coolant Inlet Temperature, deg R
C	TIPSPDG	REAL	Generator Rotor Surface Speed, ft/sec [limit is 700]
C	TOUTCLNT	REAL	Specified Generator Coolant Outlet Temperature, deg R
C	TURBTYP	CH*10	Turbine Type, either 'AXIAL' or 'RADIAL'
C	VOLTAGE	REAL	Desired Generator Output Voltage, 3Ph, line-line, RMS
C	WARNINGG	CH*64	Generator Subroutine Diagnostic Warning Message
C	WCLNT	REAL	Calculated Coolant Flow Rate, 1bm/sec

IMPLICIT REAL (A-H)  
 IMPLICIT INTEGER (I-J)  
 IMPLICIT REAL (K-Z)

CHARACTER\*10 COMPTYPE, TURBTYP, CLNTYPE, TVAR(18)

CHARACTER\*20 GENTYPE, INTTYPE  
 CHARACTER\*64 ERRORT, ERRORC, ERRORR, ERRORG, ERRORF, WARNINGT, WARNINGC,  
 & WARNINGR, WARNINGG

COMMON/DIAGNOS/ERRORT, ERRORC, ERRORR, ERRORG, ERRORF, WARNINGT,  
 & WARNINGC, WARNINGR, WARNINGG

COMMON/OPTIM/ VAR(18), TVAR, IPRINT

COMMON/CONFIG/COMPTYPE, TURBTYP, GENTYPE, INTTYPE, CLNTYPE  
 COMMON/ALTERNTR/DGENRTR, DGENSTR, LGENTOT, MASSGEN, TIPSPDG, COE,  
 & ETAGEN, COOLING, WCLNT, VOLTAGE, KVA, GENASP, TINCLNT, TOUTCLNT,  
 & CPCLNT, LIFETIME  
 COMMON/CYCLE/TEMP(17), PRESS(17), FLOW(17), PR(17), BETA, COMFLOW,  
 & PRC, EFFC, EFFT, EFFR, EFFA, XBLC, XBLT, SPEED, POWER, PWRFCTR,  
 & DPOP6, DPOP7, DPOP8, DPOP9, DPOP10, DPOP13, DPOP14, DPOP15,  
 & DPOP16, DPOP17, NETSP, GROSSEP, EFFCYCLE, WINDAGE  
 COMMON/CONST/PI, RU, GO, JCON, RZERO

DIMENSION ZERO(9)

```
EQUIVALENCE (ZERO(1),DGENRTR),(ZERO(2),DGENSTR),(ZERO(3),LGENTOT),
& (ZERO(4),MASSGEN),(ZERO(5),TIPSPDG),(ZERO(6),COE      ),
& (ZERO(7),ETAGEN ),(ZERO(8),COOLING),(ZERO(9),WCLNT )
```

```
C **** Zero-out generator variables such that zero's are returned in the
C **** event of failure to reach a solution.
```

```
DO 5, I=1,9
5 ZERO(I)=0.
```

```
C **** Compute the design generator output kVA as the basis for sizing.
C **** Reset generator error and warning messages. Identify generator type.
```

```
KVA=POWER/PWRFCTR
ERRORG=' '
WARNINGG=' '
IF (GENTYPE.EQ.' ') GENTYPE='RING WOUND TPTL PMG'
IF (INTTYPE.EQ.' ') INTTYPE='TRANSFORMER'
IF (CLNTTYPE.EQ.' ') THEN
CLNTTYPE='DOWTHERM A'
CPCLNT=.5
TINCLNT=920.
TOUTCLNT=940.
END IF
ITER=0
```

```
C **** Check the specified rotor aspect ratio [L/D]. If outside the range
C **** of 2 to 3, set equal to 2.5.
```

```
IF (GENASP.LT.2..OR.GENASP.GT.3.) THEN
GENASP=2.5
WARNINGG=
& 'SPEC'D. GENERATOR ASPECT RATIO OUT OF RANGE; RESET TO 2.5'
END IF
```

```
C **** Check the specified electrical power factor. If outside the range
C **** of .7 to 1.0, set equal to 0.9.
```

```
IF (PWRFCTR.LT.0.7.OR.PWRFCTR.GT.1.0) THEN
PWRFCTR=0.9
KVA=POWER/PWRFCTR
WARNINGG=
& 'SPEC'D. ELECTRICAL POWER FACTOR OUT OF RANGE; RESET TO 0.9'
END IF
```

```
C **** Perform an input check to assure that design parameters
```

C        are within the validity limits of the algorithms. If not, load an  
C \*\*\*\* approrpriate diagnostic message into ERRORG and exit subroutine.

```
IF (VOLTAGE.LT.1000..OR.VOLTAGE.GT.10000.) THEN
ERRORG='SPECIFIED OUTPUT VOLTAGE IS OUTSIDE VALID 1-10 KV RANGE'
RETURN
END IF
```

```
IF (KVA.LT.50..OR.KVA.GT.6000.) THEN
ERRORG=
& 'SPECIFIED POWER LEVEL IS OUTSIDE VALID 50-6000 KVA RANGE'
RETURN
END IF
```

C \*\*\*\* Compute the rotor sizing coefficient, rotor diameter and tip speed  
C        based on design algorithms. Rotor diameter includes sleeve. Starts  
C \*\*\*\* by assuming TIPSPDG=700.

TIPSPDG=700.

```
10 COE=(MAX(450.,MIN(TIPSPDG,700.))/700.)**.468*(40.65+.00066*
& VOLTAGE*(MAX(450.,MIN(TIPSPDG,700.))/700.)**2.5)*KVA**(-.075)
DGENRTR=COE*(KVA/(SPEED*GENASP))**.33333
OLDTD=TIPSPDG
TIPSPDG=PI*DGENRTR/12.*SPEED/60.
ITER=ITER+1
IF (ITER.GT.50) THEN
ERRORG='ITERATION LIMIT EXCEEDED IN SUBROUTINE GENSIZE'
WRITE (*,20) ERRORG
WRITE (61,20) ERRORG
RETURN
END IF
```

C \*\*\*\* Check for convergence of design rotor tip speed.

```
IF (ABS((TIPSPDG-OLDTS)/TIPSPDG).GT..001) GO TO 10
```

C \*\*\*\* Check to insure that the 700 fps rotor tip speed limit is not  
C        violated by the currently requested rotating speed. If the limit  
C        is exceeded, a warning diagnostic message is loaded, a new lower speed  
C        value is calculated, the rotor speed variable is reset, and processing  
C \*\*\*\* continues.

```
IF (TIPSPDG.GT.700.)THEN
WARNINGG=
& 'SPEC'D. ROTOR SPEED CAUSES GENERATOR TIP SPEED>700 FPS; N RESET'
COE=(40.65+.00066*VOLTAGE)*KVA**(-.075)
SPEED=(700.*60.*12./PI/COE)**1.5*SQRT(GENASP/KVA)
VAR(6)=SPEED
DGENRTR=COE*(KVA/(SPEED*GENASP))**.33333
TIPSPDG=PI*DGENRTR/12.*SPEED/60.
END IF
```

```
C **** Check to insure that generator surface speed is >= 450 fps to  
C      preserve accuracy in mass prediciton. Also check that the specified
```

```
C **** electrical interface is consistent with the algorithm.
```

```
IF (TIPSPDG.LT.450.) WARNINGG=  
& 'GENERATOR SURFACE SPEED BELOW 450 FPS; MASS ESTIMATE IS SUSPECT'  
IF (INTTYPE.NE.'TRANSFORMER') WARNINGG=  
&'GEN INTERFACE OTHER THAN TRANSFORMER SPECIFIED; ACCURACY SUSPECT'
```

```
C **** Estimate the OD and total length of the generator EM assembly.
```

```
DGENSTR=DGENRTR*(TIPSPDG/700.)**(-.4)*(2.14-.120*KVA**.175-  
& .225E-4*VOLTAGE)  
LGENTOT=DGENRTR*(2.98-.02*DGENRTR)*(GENASP+.48)/2.98
```

```
C **** Compute the electro-magnetic [EM] mass of the current configuration  
C Mass includes rotor magnet and sleeve, magnetic conductor, copper, and  
C **** insulation. A mass correction is also made for L/D.ne.2.5.  
C Correct for L/D values other than 2.5 by scaling mass by  
C [(GENASP+.48)/2.98] which reflects the approximate overall generator  
C length ratio.
```

```
MASSGEN=1.938*(TIPSPDG/700.)**(-.591)*(1.0467-.000033*VOLTAGE)*
```

```
& DGENRTR**2.85*(GENASP+.48)/2.98
```

```
C **** All reference design cases run at 500 and 700 fps had overall  
C efficiencies [including small windage loss] in a tightly clustered  
C group with no specific trends. Therefore, the average value of  
C **** 96.62 % is assigned to all algorithm designs.
```

```
ETAGEN=.9662
```

```
RETURN
```

```
20 FORMAT (1X,A64)  
END
```

```
SUBROUTINE STRNTH (TMATL,FPL,SIGPV)  
$DEBUG  
IMPLICIT REAL (A-H,K-Z)
```

```
C ***** DESIGN STRENGTH SUBROUTINE *****
```

```
C **** Convert to degrees K  
TT = TMATL/1.8D0
```

C TZM Correlation for 1% Creep

```
CH = -22.0356
A = -77.43
B = -2530.33
C = 39963.9
```

C \*\*\*\* Compute log10 (Lifetime in hours)

```
THET = DLOG10(FPL*8.76D3)
```

C \*\*\*\* Initialize log10 (Sigma) = 3., Sigma = 1000. MPa

```
SIGKSI=10.
```

```
DO 50 I=1,100
```

C \*\*\*\* Compute Lifetime based on current Sigma

```
THETA=CH+(A*SIGKSI+B*DLOG10(SIGKSI)+C)/TT
FUNC = THET - THETA
FPRIME = -(A+B/(SIGKSI*DLOG(10.)))/TT
DELTA = FUNC/FPRIME
SIGKSSIO = SIGKSI
SIGKSI = SIGKSI - DELTA
IF (DABS(FUNC) .LT. 1.D-6) GO TO 60
50 CONTINUE
```

C \*\*\*\* Convert from ksi to psi

```
60 SIGPV = SIGKSI*1000.
```

```
RETURN
END
```

SUBROUTINE RADTURB

C \*\*\*\* RADTURB is checked-out and final as of 9-25-92  
\$DEBUG

```
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
REAL JCON
```

```
CHARACTER*10 GAS(2),COMPTYPE,TURBTYPE,CLNTTYPE,TVAR(18)
CHARACTER*20 GENTYPE,INTTYPE
CHARACTER*64 ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,WARNIGC,
& WARNIGR,WARNIGG
```

```
COMMON/OPTIM/ VAR(18),TVAR,IPRINT
```

```
COMMON/DIAGNOS/ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,
```

& WARNINGC,WARNINGR,WARNINGG

```

COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
& RADNC,ANSQC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,
& RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT
COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,
& CPCLNT,LIFETIME
COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE
COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLOW,
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,
& DPOP16,DPOP17,NETSP,GROSSE,PFCYC,WINDAGE
COMMON/CONST/PI,RU,GO,JCON,RZERO

```

C \*\*\*\* Nomenclature for Subroutine RADTURB

C ACRST1	REAL	Turbine inlet critical sonic velocity, ft/sec
C ACRST2	REAL	Turbine rotor exit critical sonic velocity, ft/sec
C ACRST3	REAL	Turbine diffuser exit critical sonic velocity, ft/sec
C ANG1	REAL	Flow angle at nozzle exit, degrees from radial
C ANG2	REAL	Flow angle at rotor tip, degrees from radial
C ARAT	REAL	Turbine diffuser area ratio, dmls
C AREA2	REAL	Annular area required at rotor exit, sq in
C AMAX	REAL	Max annular area at rot.exit w/ current ETTR, PHI, dmls
C B	REAL	Rotor inlet blade height [B-width], inches
C C2	REAL	Sonic velocity at rotor exit, R
C CLEAROB	REAL	Clearance:B-width ratio, dmls
C DELHACT	REAL	Actual delta enthalpy across turbine, BTU/lbm
C DELHISEN	REAL	Isentropic delta enthalpy across turbine, BTU/lbm
C DELTACT	REAL	Actual delta total temperature across turbine, R
C DELTISEN	REAL	Isentropic delta total temperature across turbine, R
C DENRAT	REAL	Static:total density ratio at rotor inlet, dmls
C EBASE	REAL	Base [uncorrected] overall adiabatic efficiency, dmls
C EFFT	REAL	Turbine t-t adiabatic efficiency incl. diffuser, dmls
C EPOLY	REAL	Polytropic efficiency from Balje reference, dmls
C ERRORT	CHAR*64	Turbine design error message
C ETAC	REAL	Turbine t-s ad. efficiency after clearance corr., dmls
C ETAR	REAL	Turbine t-s ad. efficiency after Re No. correction., dmls
C ETATTS	REAL	Turbine t-s ad. efficiency after disc friction corr, dmls
C ETATURB	REAL	Turbine t-t ad. efficiency after conversion, dmls
C ETTR	REAL	Diffuser OD to rotor tip OD ratio, dmls
C EXDIA	REAL	Exducer shroud diameter, inches
C HUBDIA	REAL	Exducer hub diameter, inches
C ITER	INTEG	Iteration counter
C MACH2	REAL	Rotor exit Mach No., dmls
C MACH3	REAL	Diffuser exit Mach No., dmls
C MUT	REAL	Turbine average fluid viscosity, lbm/ft-sec
C NST	REAL	Turbine specific speed [Balje form], dmls
C NSTURB	REAL	Turbine specific speed, rpm/ft^3/4-sec^1/2
C PHI	REAL	Exducer hub:tip radius ratio, dmls

C	PSx	REAL	Static pressure at station "x", psia
C	PSOPT2	REAL	Static:total pressure ratio at rotor exit, dmls
C	PSOPT3	REAL	Static:total pressure ratio at diffuser exit, dmls
C	PTx	REAL	Total pressure at station "x", psia
C	R	REAL	Working fluid gas constant, ft-lbf/lbm-R
C	REFFT	REAL	Turbine Reynolds No., dmls
C	REX	REAL	Exducer shroud radius, inches
C	RHO1	REAL	Rotor inlet fluid [total conditions] density, 1bm/cu ft
C	RHO2	REAL	Static density at rotor exit, 1bm/cu ft
C	RHUB	REAL	Exducer hub radius, inches
C	TSx	REAL	Static temperature at station "x", R
C	TSOTT2	REAL	Static:total temperature ratio at rotor exit, dmls
C	TSOTT3	REAL	Static:total temperature ratio at diffuser exit, dmls
C	TTx	REAL	Total temperature at station "x", R
C	TTD	REAL	Turbine discharge total temperature, R
C	TURBDIA	REAL	Turbine rotor tip diameter, inches
C	TURBSS	REAL	Turbine specific speed, rpm/ft <sup>3/4</sup> -sec <sup>1/2</sup>
C	U1	REAL	Turbine rotor tip speed, ft/sec
C	U2H	REAL	Exducer hub tangential velocity, ft/sec
C	U2T	REAL	Exducer tip tangential velocity, ft/sec
C	UTIPT	REAL	Rotor tips speed, ft/sec
C	VACR	REAL	Critical velocity ratio [V/Acr] at rotor inlet, dmls
C	VU	REAL	Tangential fluid velocity at rotor inlet, ft/sec
C	VX	REAL	Meridinal fluid velocity at rotor inlet, ft/sec
C	VX2	REAL	Meridinal fluid velocity at rotor exit, ft/sec
C	W1	REAL	Blade relative velocity at rotor inlet, ft/sec
C	W2H	REAL	Blade relative velocity at rotor exit hub, dmls
C	W2T	REAL	Blade relative velocity at rotor exit tip, dmls
C	WARNINGT	CHAR*64	Turbine design warning message
C	WTAPT2	REAL	Flow parameter. [=f(MACH2)], 1bm-R <sup>1/2</sup> /s-lbf
C	Z	REAL	Blade + splitter count at rotor inlet, dmls

C \*\*\*\* Initialize and equate turbine stage design variables; reset counters.

```
ITER=0
ERRQRT=' '
WARNINGT=' '
```

C \*\*\*\* Determine the average fluid viscosity in the turbine stage.

```
MUT=AVVISC(TEMP(10),TEMP(12))
```

C \*\*\*\* Assume the Diffuser Area Ratio is 2.65 [as in TM X-2357]

```
ARAT=2.65
MACH2=.2
R=RU/MOLWT
ACRST1=SQRT((2*GAMMA/(GAMMA+1))*G0*R*TEMP(10))
Z=20.
ANG1=72.
```

C \*\*\*\* Begin analysis assuming exducer tip:wheel tip ratio = 0.7

ETTR=.7  
PHI=.4  
ANG2=70.  
ETATTS=.88

C \*\*\*\* Initialize PS2, Rotor Exit Static Pressure

PSOPT2=.97  
PSOPT3=.99  
TSOTT2=.99  
TSOTT3=1.  
PS2=PRESS(12)\*.95  
PT2=PS2/PSOPT2  
PT3=PRESS(12)  
PS3=PT3\*PSOPT3

C \*\*\*\* Compute the turbine discharge temperature and begin the design

C \*\*\*\* iteration to converge efficiency.

10 PS2OLD=PS2  
PS2=PT2\*PSOPT2  
DELTISEN=TEMP(10)\*(1.-(PS3/PRESS(10))\*\*((GAMMA-1)/GAMMA))  
DELHISEN=CP\*DELTISEN  
DELTACT=DELTISEN\*ETATTS  
DELHACT=DELHISEN\*ETATTS  
TTD=TEMP(10)-DELTACT  
TTDISEN=TEMP(10)-DELTISEN  
TS2=TTD\*TSOTT2  
TS3=TTD\*TSOTT3  
ACRST2=SQRT((2\*GAMMA/(GAMMA+1))\*GO\*R\*TTD)  
ACRST3=ACRST2  
U1=SQRT(GO\*JCON\*DELHACT/(1.-2./Z))

C \*\*\*\* Compute wheel diameter, base efficiency, Reynolds' No. & correction.

C \*\*\*\* Turbine efficiency based on Balje Reference

TURBDIA=U1\*60.\*12./(SPEED\*PI)  
NST=PI/30.\*SPEED\*SQRT(FLOW(10)\*R\*TTDISEN/(144.\*PS2))/  
& (DELHISEN\*JCON\*GO)\*\*.75  
NSTURB=SPEED\*SQRT(FLOW(10)\*R\*TS3/(144.\*PS2))/  
& (DELHACT\*JCON)\*\*.75

C \*\*\*\* Compute radial turbine uncorrected polytropic efficiency per Balje

C \*\*\*\* Curve fit of data taken from:

C           Balje, O. J., "Turbomachines: A Guide to Design, Selection,  
C           and Theory", John Wiley and Sons, Inc., New York, 1981.

EPOLY=.64865+.71927\*(ALOG10(NST)+1.) -1.9947\*(ALOG10(NST)+1.)\*\*2+  
& 5.6333\*(ALOG10(NST)+1.)\*\*3-6.6197\*(ALOG10(NST)+1.)\*\*4+  
& 2.2954\*(ALOG10(NST)+1.)\*\*5

```
YT=1-(PS3/PRESS(10))**((GAMMA-1.)/GAMMA)
ZT=EPOLY*ALOG10((PS3/PRESS(10))**((GAMMA-1.)/GAMMA))
EBASE=(1.-10.**ZT)/YT
```

20 CONTINUE

```
RHO1=144.*PRESS(10)/(R*TEMP(10))
VU=(1.-2/Z)*U1
VX=VU/TAN(ANG1*PI/180)
VACR=VU/(ACRST1*SIN(ANG1*PI/180))
```

C \*\*\*\* Compute rotor relative velocity and rotor 'B-width'.

```
W1=SQRT((VX**2)+(VU-U1)** 2)
DENRAT=(1.+((GAMMA-1.)/(GAMMA+1.))*VACR**2)**(1./(1.-GAMMA))
B=144*FLOW(10)/(DENRAT*VX*TURBDIA*PI*RHO1)
```

C \*\*\*\* Define the exducer geometry and flow vectors.

30 REX=(ETTR\*TURBDIA)/2.

```
EXDIA=REX*2.
U2T=ETTR*U1
VX2=U2T/TAN(ANG2*PI/180.)
```

M2OLD=0.

C DO WHILE (ABS(M2OLD-MACH2).GT..0001)

```
32 M2OLD=MACH2
ICODE=1
CALL GASDYN (ICODE,MACH2,PSOPT2,TSOTT2,VOCT2,ROR2,AOAST2,
& WTAP2,WTAPS2,QOP2)
C2=SQRT(GAMMA*GO*R*TS2)
MACH2=VX2/C2
IF (ABS(M2OLD-MACH2).GT..0001) GOTO 32
RHO2=144.*PT2/(R*TTD)*ROR2
```

C \*\*\*\* Compute Reynolds No Correction

```
REFFT=RHO2*SPEED*(TURBDIA/12.)**2*PI/(30.*MUT)
IF (REFFT.GT.4.E6) THEN
ETAR=1.-(1.-EBASE)*(2.E6/4.E6)**.2
ELSE
ETAR=1.-(1.-EBASE)*(2.E6/REFFT)**.2
END IF
```

```
AMAX=PI/4.*(TURBDIA*ETTR)**2*(1.-4**2)
AREA2=FLOW(10)*SQRT(TTD)/PT2/WTAP2
```

IF (AREA2.GT.AMAX) GO TO 40

```
RHUB=SQRT(REX**2-AREA2/PI)
HUBDIA=2.*RHUB
GO TO 50
```

```

40 ETTR=ETTR+.01
  IF (ETTR.GT..9) THEN
    ERRORT=
    & ' RADIAL TURBINE GEOMETRY NOT POSSIBLE; SHROUT/TIP RATIO > 0.9'
    ETATURB=.6

    RETURN
  ELSE
    GO TO 30
  END IF

50 AREA2=PI*(REX**2-RHUB**2)
  PHI=RHUB/REX
  W2T=SQRT(VX2**2+U2T**2)
  U2H=PHI*U2T
  W2H=SQRT(VX**2+U2H**2)
  IF ((W2H/W1).LT.1.05) THEN
    WARNINGT='FLOW DIFFUSION/SEPARATION PREDICTED IN RADIAL TURBINE'

435 FORMAT (1X,A64)
  ETATURB=.6

  RETURN
END IF

C **** Estimate the required running clearances and their contribution
C **** to stage efficiency.

CLEAROB=MAX(.02,.01/B)
ETAC=ETAR*(1.0078-.39*CLEAROB)
ETAOLD=ETATTS
ETATTS=ETAC

C **** Assume the diffuser effectiveness [per NASA TM X-2357] is 0.65.

MACH3=MACH2/ARAT
AREA3=AREA2*ARAT

M30LD=0.
c DO WHILE (ABS(M30LD-MACH3).GT..001)
52 PT2=PS3/(.65+.35*PSOPT2)
  PS2=PT2*PSOPT2
  ICODE=1
  CALL GASDYN (ICODE,MACH3,PSOPT3,TSOTT3,VOCT3,ROR3,AOAST3,
& WTAPT3,WTAPS3,QOP3)
  PS3=PT3*PSOPT3
  WDOT=WTAPT3*AREA3*PT3/SQRT(TTD)
  M30LD=MACH3
  MACH3=FLOW(10)/WDOT*MACH3
  IF (ABS(M30LD-MACH3).GT..001) GOTO 52

  ITER=ITER+1

```

```

IF (ITER.GT.200) THEN
  ERRORT=' ITERATION LIMIT EXCEEDED IN RADIAL TURBINE ROUTINE'
  WRITE (61,435) ERRORT
  WRITE (*,435) ERRORT
  ETATURB=.6

  RETURN
END IF

C **** Test for convergence of efficiency prediction and diffuser loss.

  IF (ABS((PS2OLD-PS2)/PS2).GT..00001) GO TO 10
  IF (ABS((ETATTS-ETAOLD)/ETAOLD).GT..00001) GO TO 10

  TEMP(12)=TTD
  UTIPT=U1
  IF (UTIPT.GT.1800.) WARNINGT='TURBINE TIP SPEED HIGH - CREEP/STRES
& S RUPTURE A CONCERN'
  IF (UTIPT.GT.2000.) WARNINGT='TURBINE TIP SPEED GREATER THAN 2000
& FPS NOT ADVISED'
  TURBSS=NSTURB

C **** Convert from t-s to t-t overall efficiency
  ETATURB=ETATTS*(1.-(PS3/PRESS(10))**((GAMMA-1)/GAMMA))/(
  & (1.-(PT3/PRESS(10))**((GAMMA-1)/GAMMA))
  ETATURB=MAX(0.6,MIN(.98,ETATURB))

C **** Compute approximate rotor stress levels in psi, TZM alloy
  RHOTZM=.37
C **** Approximate centrif. stress based on BRU; thermal ss stress assumed
C **** to "creep relax".
  SIGMA=.03*UTIPT**2*RHOTZM/.29
C **** Using the rotor bore abs. temperature ratio from BRU analyses
  TMATL=.37*TEMP(10)+.63*TTD
  CALL STRNTH (TMATL,LIFETIME,SIGPV)
  IF (SIGMA.GT.SIGPV) ERRORT='RADIAL TURBINE OPERATING ABOVE BORE ST
& RESS LIMIT; RED. TIT OR INC: MW'

  IF (IPRINT.GE.4) THEN
    WRITE (61,100) PRESS(10),PS2,PT2,PS3,PT3,NST
    WRITE (61,110) TEMP(10),TS2,TS3,TTD,B,REFFT
    WRITE (61,120) TURBDIA,EXDIA,HUBDIA
    WRITE (61,130) MACH2,WTAP2,MACH3,WTAP3
    WRITE (61,140) EPOLY,EBASE,ETAR,ETAC,ETATTS,ETATURB
  END IF

  RETURN

100 FORMAT (' PRESS(10),PS2,PT2,PS3,PT3,NST'          '/1X,7F10.3)
110 FORMAT (' TEMP(10),TS2,TS3,TTD,B,REFFT'          '/1X,5F10.3,
  & 2E10.3)
120 FORMAT (' TURBDIA,EXDIA,HUBDIA'                 '/1X,7F10.3)

```

```
130 FORMAT (' MACH2,WTAPT2,MACH3,WTAPT3      '/1X,7F10.6)
140 FORMAT (' EPOLY,Ebase,ETAR,ETAC,ETATTS,ETATURB   '/1X,7F10.6)
END
```

```
SUBROUTINE GASDYN (Icode,M,PSOPT,TSOTT,VOCT,RHOORHO,AOAST,
& WRTOAPS,WRTOAPT,QOP)
```

```
C **** GASDYN is checked-out and final as of 10-06-92
$DEBUG
```

```
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
CHARACTER*10 GAS(2)
REAL JCON
```

```
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
COMMON/CONST/PI,RU,GO,JCON,RZERO
```

```
C **** Icode designates which information is provided and what is to
C **** be calculated:
```

```
C           Icode=1 means Mach No. is specified
C           Icode=2 means WT^.5/Ap is specified
```

```
IF(ICODE.EQ.2) THEN
IF(WRTOAPT.LE.0.)RETURN
M=SQRT(-1./(GAMMA-1.))+SQRT(1./(GAMMA-1.)**2+2.*RU/MOLWT/
& (GAMMA*(GAMMA-1.)*GO)*WRTOAPS**2))
END IF
```

```
PSOPT=(1.+(GAMMA-1.)/2.*M**2)**(-GAMMA/(GAMMA-1.))
TSOTT=1./(1.+(GAMMA-1.)/2.*M**2)
```

```
VOCT=M/SQRT(1.+(GAMMA-1.)/2.*M**2)
RHOORHO=(1.+(GAMMA-1.)/2.*M**2)**(-1./(GAMMA-1.))
AOAST=1./M*((1.+(GAMMA-1.)/2.*M**2)/((GAMMA+1.)/2.))
& **((GAMMA+1.)/(2.*(GAMMA-1.)))
WRTOAPS=SQRT(GAMMA*GO/(RU/MOLWT))*M*SQRT(1.+(GAMMA-1.)/2.*M**2)
WRTOAPT=SQRT(GAMMA*GO/(RU/MOLWT))*M*(1.+(GAMMA-1.)/2.*M**2)
& **(-(GAMMA+1.)/(2.*(GAMMA-1.)))
QOP=GAMMA/2.*M**2*(1+(GAMMA-1.)/2.*M**2)**(-GAMMA/(GAMMA-1.))
```

```
RETURN
END
```

```
SUBROUTINE RADCOMP
```

```
C **** RADCOMP is checked-out and final as of 10-06-92
$DEBUG
```

```
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
```

REAL JCON

CHARACTER\*10 GAS(2), COMPTYPE, TURBTYPE, CLNTTYPE, TVAR(18)

CHARACTER\*20 GENTYPE, INTTYPE

CHARACTER\*64 ERRORT, ERRORC, ERRORR, ERRORG, ERRORF, WARNINGT, WARNINGC,  
& WARNINGR, WARNINGG

COMMON/OPTIM/ VAR(18), TVAR, IPRINT

COMMON/DIAGNOS/ERRORT, ERRORC, ERRORR, ERRORG, ERRORF, WARNINGT,  
& WARNINGC, WARNINGR, WARNINGG

COMMON/CONFIG/COMPTYPE, TURBTYPE, GENTYPE, INTTYPE, CLNTTYPE  
COMMON/FLUID/ GAS, GAMMA, MOLWT, CP, PRNDL  
COMMON/CYCLE/TEMP(17), PRESS(17), FLOW(17), PR(17), BETA, COMFLOW,  
& PRC, EFFC, EFFT, EFFR, EFFA, XBLC, XBLT, SPEED, POWER, PWRFCTR,  
& DPOP6, DPOP7, DPOP8, DPOP9, DPOP10, DPOP13, DPOP14, DPOP15,  
& DPOP16, DPOP17, NETSP, GROSSEP, EFFCYCLE, WINDAGE  
COMMON/AERODYN/ETACOMP, COMPDIA, UTIPC, RADHC, RAD1C,  
& RADNC, ANSQC, ANSQCL, COMPSS, ISTGC, ETATURB, TURBDIA, UTIPT,  
& RADHT, RADIT, RADNT, ANSQTL, TURBSS, ISTGT

COMMON/CONST/PI, RU, GO, JCON, RZERO

C \*\*\*\* Nomenclature for Subroutine RADCOMP

C ACR2	REAL	Fluid critical sonic velocity at rotor exit, ft/sec
C B	REAL	Rotor tip blade height [B-widht], inches
C CDP	REAL	Compressor discharge total pressure, psia
C CDT	REAL	Compressor discharge total temperature, R
C CLEAROB	REAL	Compressor clearance:B-with ratio, dmls
C COMPDIA	REAL	Compressor rotor tip diameter, inches
C DTOT	REAL	Compressor t-t delta T/T, dmls
C EAD	REAL	Compressor uncorrected adiabatic efficiency, dmls
C EPOLY	REAL	Comp. uncorrected polytropic efficiency [Balje], dmls
C ERRORC	CHAR*64	Compressor design error message
C ETAAD	REAL	Compressor adiabatic efficiency w/ Re correction, dmls
C ETACOMP	REAL	Compressor adiabatic efficiency w/ all corrections, dmls
C FCW	REAL	Compressor work factor [Cu/U], dmls
C H1	REAL	Compressor inlet flow enthalpy, BTU/lbm
C H2	REAL	Compressor discharge flow enthalpy, BTU/lbm
C ITER	INTEG	Iteration counter
C M2STAR	REAL	Critical Mach No. at rotor exit, dmls
C MUC	REAL	Viscosity at compressor inlet, lbm/ft-sec
C NSC	REAL	Compressor inlet specific speed, dmls
C PS2	REAL	Rotor exit static pressure, psia
C PT2	REAL	Rotor exit total pressure, psia
C PSOPT2	REAL	Static:total pressure ratio at rotor exit, dmls

C	Q1	REAL	Rotor inlet volume flow, cu ft/sec
C	Q2	REAL	Rotor discharge volume flow, cu ft/sec
C	REC	REAL	Compressor Reynolds Number, dmls
C	RGAS	REAL	Gas constant for working fluid, ft-lbf/lbm-R
C	RHO1	REAL	Compressor inlet total fluid density, lbm/cu ft
C	RHO2	REAL	Compressor discharge fluid density, lbm/cu ft
C	TS2	REAL	Rotor exit static temperature, R
C	UTIPC	REAL	Compressor rotor tip speed, ft/sec
C	VR2	REAL	Fluid radial velocity at rotor exit, ft/sec
C	VU2	REAL	Fluid tangential velocity at rotor exit, ft/sec
C	V2	REAL	Fluid velocity at rotor exit, ft/sec
C	WARNINGC	CHAR*64	Compressor design warning message
C	XSS	REAL	Compressor mean specific speed, rpm/ft^3/4-sec^1/2

C \*\*\*\*\* INPUTS: PRAT,SPEED,P1,T1,COMFLOW,MOLWT

```

ITER=0
WARNINGC=' '
ERRORC=' '

MUC=VISCOS(TEMP(1))
RGAS=RU/MOLWT
FCW=.7
CDP=PRESS(1)*PRC
PSOPT3=.98
PT3=PRESS(4)
ETACOMP=.85
ETACTS=.85

```

```

10 DTOT=((PRC*PSOPT3)**((GAMMA-1.)/GAMMA)-1.)/ETACTS
COMPDIA=SQRT(DTOT*TEMP(1)/FCW*GO*JCON*CP)/(PI*SPEED/12./60.)
CDT=((PRC*PSOPT3)**((GAMMA-1.)/GAMMA)-1.)/ETACTS+1.)*TEMP(1)
H1=TEMP(1) * CP
H2=CDT * CP

```

```

Q1=FLOW(1)*TEMP(1)*RU/PRESS(1)/MOLWT/144.
Q2=FLOW(1)*TEMP(4)*RU/PRESS(4)/MOLWT/144.
XSS=SPEED*(Q1*Q2)**.25/(788.16*(H2-H1))**.75
NSC=PI/30.*SPEED*SQRT(Q1)/(GO*JCON*(H2-H1)*ETACTS)**.75

```

C \*\*\*\* Compute radial compressor uncorrected polytropic efficiency per Balje  
C \*\*\*\* Curve fit of data taken from:  
C       Balje, O. J., "Turbomachines: A Guide to Design, Selection,  
C       and Theory", John Wiley and Sons, Inc., New York, 1981.

```

EPOLY=.38186+.40860*(ALOG10(NSC)+1.)+1.1734*(ALOG10(NSC)+1.)**2-
& 1.5643*(ALOG10(NSC)+1.)**3+.47413*(ALOG10(NSC)+1.)**4

```

```

YC=(PRC*PSOPT3)**((GAMMA-1.)/GAMMA)-1.
ZC= ALOG10((PRC*PSOPT3)**((GAMMA-1.)/GAMMA))/EPOLY

```

EAD=YC/(10.\*\*ZC-1.)

C \*\*\*\* Correct Compressor Efficiency for Reynolds No. based on ASME  
C \*\*\*\* Code PTC-10

RHO1=PRESS(1)\*144./(RGAS\*TEMP(1))  
REC=RHO1\*SPEED\*(COMPDI/12.)\*\*2\*PI/(30.\*MUC)  
IF (REC.LT.4.E6) THEN

ETAAD=1.+(EAD-1.)\*(2.E6/REC)\*\*.1  
ELSE  
ETAAD=1.+(EAD-1.)\*(2.E6/4.E6)\*\*.1  
END IF

C \*\*\*\* Correct compressor efficiency for clearance losses  
C \*\*\*\* Assume .7 pressure recovery in the compressor diffuser

UTIPC=COMPDI/12.\*PI\*SPEED/60.  
VU2=.7\*UTIPC  
C Assume Alpha2 = 75 deg  
VR2=VU2/TAN(75.\*PI/180.)  
ACR2=SQRT(2.\*GAMMA/(GAMMA+1.)\*GO\*RGAS\*CDT)  
V2=VU2/SIN(75.\*PI/180.)  
M2STAR=V2/ACR2  
PSOPT2=(1.-(GAMMA-1.)/(GAMMA+1.)\*M2STAR\*\*2)\*\*(GAMMA/(GAMMA+1.))  
PT2=CDP/(.7\*(1.-PSOPT2)+PSOPT2)  
PS2=PT2\*PSOPT2  
TS2=CDT\*(1.-(GAMMA-1.)/(GAMMA+1.)\*M2STAR\*\*2)  
RHO2=PS2\*144./(RGAS\*TS2)  
B=144.\*FLOW(1)/(RHO2\*VR2\*PI\*COMPDI)  
CLEAROB=MAX(.02,.01/B)

C \*\*\*\* Assume a 3:1 ratio of M2STAR/M3STAR  
PSOPT3=(1.-(GAMMA-1.)/(GAMMA+1.)\*(M2STAR/3.)\*\*2)\*\*(GAMMA/  
& (GAMMA+1.))

C WRITE(61,234)UTIPC,VU2,VR2,ACR2,V2,M2STAR,PSOPT2,PT2,PS2,TS2,RHO2,  
C & B,CLEAROB,FLOW(1)  
C 234 FORMAT(' UTIPC,VU2,VR2,ACR2,V2,M2STAR,PSOPT2,PT2,PS2,TS2,RHO2,B,CL  
C &EAROB,FLOW(1)'/3(1X,5E12.5/))  
C WRITE(61,345)NSC,EPOLY,EAD,REC,ETAAD,PSOPT3  
C 345 FORMAT (' NSC,EPOLY,EAD,REC,ETAAD,ETACTS,PSOPT3'2(/1X,5E12.5))

C \*\*\*\* Compute efficiency correction for clearance loss per Balje  
C \*\*\*\* Curve fit of data taken from:  
C Balje, O. J., "Turbomachines: A Guide to Design, Selection,  
and Theory", John Wiley and Sons, Inc., New York, 1981.

ETACTS=ETAAD\*(1.01848-.924\*CLEAROB)

IF (ABS(ETACTS - ECOLD).LT..00001) THEN

UTIPC=PI\*COMPDIA/12.\*SPEED/60.

C \*\*\*\* Compressor tip speed limit per GFSD experience

IF (UTIPC.GT.1800.) WARNINGC='RADIAL COMPRESSOR TIP SPEED EXCEEDS  
&1800 FPS LIMIT'

C \*\*\*\* Convert efficiency to total-to-total

ETACOMP=ETACTS\*(PRC\*\*((GAMMA-1.)/GAMMA)-1.)/((PRC\*PSOPT3)\*\*

& ((GAMMA-1.)/GAMMA)-1.)

ETACOMP=MAX(0.6,MIN(.95,ETACOMP))

RETURN

ELSE

ECOLD = ETACTS

ITER=ITER+1

IF (ITER.GT.50) THEN

ERRORC='ITERTION LIMIT EXCEEDED IN RADIAL COMPRESSOR ROUTINE'

ETACOMP=.6

RETURN

END IF

GOTO 10

END IF

END

#### SUBROUTINE AXCOMP

C \*\*\*\* AXCOMP is cheked-out and final as of 10-06-92

\$DEBUG

IMPLICIT REAL (A-H)

IMPLICIT INTEGER (I-J)

IMPLICIT REAL (K-Z)

REAL JCON

CHARACTER\*10 GAS(2),COMPTYPE,TURBTYPE,CLNTTYPE,TVAR(18)

CHARACTER\*20 GENTYPE,INTTYPE

CHARACTER\*64 ERRORT,ERRORC,ERRORT,ERRORG,ERRORF,WARNINGT,WARNINGC,  
& WARNINGR,WARNINGG

COMMON /OPTIM/VAR(18),TVAR,IPRINT

COMMON/DIAGNOS/ERRORT,ERRORC,ERRORT,ERRORG,ERRORF,WARNINGT,  
& WARNINGC,WARNINGR,WARNINGG

COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE

COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDL

COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,  
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,  
& CPCLNT,LIFETIME

COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLOW,  
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,  
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,  
& DPOP16,DPOP17,NETSP,GROSSEP,EFFCYCLE,WINDAGE

COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RADIC,  
 & RADNC,ANSQC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,  
 & RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT

COMMON/CONST/PI,RU,GO,JCON,RZERO

C \*\*\*\* Nomenclature for SUBROUTINE AXCOMP

C A1	REAL	Flow area at compressor inlet, sq in
C A2	REAL	Flow area at last stage exit, sq in
C ACR1	REAL	Compressor inlet critical velocity, ft/sec
C ACR2	REAL	Compressor last stage exit critical velocity, ft/sec
C ANSQC	REAL	Compressor flow area x speed^2 parameter, sq in-rpm^2
C ANSQCL	REAL	Compressor AN^2 limit, sq in-rpm^2
C C1	REAL	Sonic velocity at compressor inlet, ft/sec
C CR	REAL	Blade tip clearance, inches
C DELHACT	REAL	Actual compressor enthalpy rise, BTU/lbm
C DHIDEAL	REAL	Ideal compressor enthalpy rise, BTU/lbm
C ERRORC	CHAR*64	Compressor diagnostic error message
C ETACOMP	REAL	Compressor t-t adiabatic efficiency, overall, dmls
C ETACTS	REAL	Compressor t-s adiabatic efficiency [various], dmls
C ETAOLD	REAL	Previous t-s adiabatic efficiency, dmls
C ETAPOLY	REAL	Overall compressor polytropic efficiency, dmls
C ISTGC	INTEG	Compressor stage count
C ITER	INTEG	Iteration counter
C MREL	REAL	Blade relative Mach Number, dmls
C NS	REAL	Compressor specific speed, dmls
C PS1	REAL	Compressor inlet static pressure, psia
C PT1	REAL	Compressor inlet total pressure, psia
C PS2	REAL	Compressor last stage exit static pressure, psia
C PT2	REAL	Compressor last stage exit total pressure, psia
C PS3	REAL	Compressor discharge static pressure, psia
C PS3OLD	REAL	Previous value of PS3, psia
C PT3	REAL	Compressor discharge total pressure, psia
C PSOPT1	REAL	Static:total pressure ratio at compressor inlet, dmls
C PSOPT2	REAL	Static:total pressure ratio at last stage exit, dmls
C PSOPT3	REAL	Static:total pressure ratio at compressor discharge, dmls
C Q1	REAL	Compressor inlet volume flow, cu ft/sec
C QOPT2	REAL	Last stage exit dynamic head fraction of PT2, dmls
C RAD1C	REAL	Blade tip radius at compressor inlet, inches
C RADHC	REAL	Hub radius [flowpath inside radius], inches
C RADNC	REAL	Blade tip radius at compressor at last stage, inches
C REC	REAL	Compressor Reynolds Number, dmls
C RHO1	REAL	Compressor inlet density, 1bm/cu ft
C RHO2	REAL	Compressor last stage exit density, 1bm/cu ft
C SOH	REAL	Tip clearance to blade height ratio, dmls
C TS1	REAL	Compressor inlet static temperature, deg R
C TT1	REAL	Compressor inlet total temperature, deg R
C TS2	REAL	Compressor last stage exit static temperature, deg R
C TT2	REAL	Compressor last stage exit total temperature, deg R
C UHUBC	REAL	Hub velocity, ft/sec
C VOACR1	REAL	Inlet velocity:critical velocity ratio, dmls

```
C VX      REAL    Axial velocity, ft/sec
C WARNINGC CHAR*64  Compressor diagnostic warning message
C WDOT     REAL    Gas flowrate, lbm/sec
C WT1      REAL    Blade relative velocity at compressor inlet, ft/sec
```

```
C ***** INPUTS: PRAT,SPEED,PT1,TT1,FLOW,MOLWT
```

```
PT1=PRESS(1)
PSOPT1=.98
PS1=PT1*PSOPT1
TT1=TEMP(1)
TS1=TT1*PSOPT1**((GAMMA-1)/GAMMA)
PT3=PT1*PRC
PT2=PT3
PSOPT2=.97
PS2=PT2*PSOPT2
PSOPT3=.99
PS3=PT3*PSOPT3
WDOT=FLOW(1)
ETACOMP=.85
ETACTS=.85
```

```
C **** SUBROUTINE INPUTS
```

```
C ****      PT1-INLET TOTAL PRESSURE, PSIA
C ****      TT1-INLET TOTAL TEMPERATURE, R
C ****      WDOT-MASS FLOWRATE, LBM/S
C ****      N-TURBINE SPEED, RPM
C ****      PS3-DIFFUSER EXIT STATIC PRESSURE, PSIA
C ****      GAMMA-RATIO OF SPECIFIC HEATS
C ****      RGAS-WORKING FLUID GAS CONSTANT, FT-LBF/LBM-R
C ****      CP-WORKING FLUID SPECIFIC HEAT, BTU/LBM-R
C ****      CR-CLEARANCE RATIO
```

```
RGAS=RU/MOLWT
```

```
C **** Start calculations assuming a 1 stage compressor
```

```
ISTGC=1
ANSQC=0
RH01=PT1*144./RGAS/TT1
Q1=WDOT/RH01
ACR1=SQRT(2.*GAMMA/(GAMMA+1.)*GO*RGAS*TT1)
```

```
5 ITER=0
ETACTS=.85
```

```
10 DHIDEAL=TT1*CP*((PS3/PT1)**((GAMMA-1.)/GAMMA)-1.)
DELHACT=DHIDEAL/ETACTS
ERRORC=' '
WARNINGC=' '
```

```
C **** Calculate 1st Stage hub stator exit Mach Number (Sta. 1a)
```

```

UHUBC=SQRT(G0*JCON*DELHACT/(.45*FLOAT(ISTGC)))
VX=.55*UHUBC
VOACR1=VX/ACR1

C **** Compute inlet gometry
RHO1=PT1*144./RGAS/TT1*(1.-(GAMMA-1.)/(GAMMA+1.)*(VOACR1)**2)
& **(1./(GAMMA-1.))
A1=WDOT/RHO1/VX*144.

C **** Compute hub radius
RADHC=12.*UHUBC/(SPEED*PI/30)
RAD1C=SQRT(A1/PI+RADHC**2)

C **** Test for RADHC/RTIP < .6, If so, Load error message
IF ((RADHC/RAD1C).LT.0.6) ERRORC='RADHC/RAD1C < .6'

C **** Compute relative Mach Number at blade tip
TS1=TT1*(1.-(GAMMA-1.)/(GAMMA+1.)*VOACR1)**2
C1=SQRT(GAMMA*G0*RGAS*TS1)
WT1=SQRT(VX**2+(UHUBC*RAD1C/RADHC)**2)
MREL=WT1/C1

C **** If MREL > .8 load error message
IF (MREL.GT.0.8) ERRORC='MREL > .8'

C **** Compute last stage exit conditions
TT2=TT1+DELHACT/CP
ACR2=SQRT(2.*GAMMA/(GAMMA+1.)*G0*RGAS*TT2)

C **** Compute exit static:total pressure ratio
PSOPT2=(1.-(GAMMA-1.)/(GAMMA+1.)*(VX/ACR2)**2)**(GAMMA/
& (GAMMA-1.))

C **** Assume 2:1 diffusion from stator exit to duct
PSOPT3=PSOPT2**.25
PS3OLD=PS3
PS3=PT3*PSOPT3

C **** Compute exit kinetic energy
QOPT2=1.-PSOPT2

C **** Compute rotor exit pressures based on .4 diffuser coefficient of
C **** press recovery

```

```

PT2=PS3/(PSOPT2+.4*QOPT2)
PS2OLD=PS2
PS2=PT2*PSOPT2

C ***** Evaluate last stage geometry and exit velocity
RH02=144.*PT2/RGAS/TT2*(1.-(GAMMA-1.)/(GAMMA+1.)*(VX/ACR2)**2)
& **(1./(GAMMA-1.))

C ***** Compute exit flow area
A2=WDOT/RH02/VX*144.

C ***** Compute tip radius at compressor exit
RADNC=SQRT(A2/PI+RADHC**2)

C ***** Test for RADHC/RADNC > .95, If so, load error message
IF ((RADHC/RADNC).GT.0.95) ERRORC='RADHC/RADNC > .95'

C ***** Estimate Compressor Efficiencies - Polyropic value good for specific
C ***** speed range from 1.8 to 3.5
NS=SPEED*PI/30.*SQRT(WDOT/RH01)/(GO*JCON*DIDEAL/FLOAT(ISTGC))**.75
ETAOLD=ETACTS

C ***** Compute axial turbine uncorrected polytropic efficiency
C ***** Curve fit of data taken from:
C           Balje, O. J., "Turbomachines: A Guide to Design, Selection,
C           and Theory", John Wiley and Sons, Inc., New York, 1981.
C           and expanded based on GFSD analyses done in support of classified
C           programs.
ETAPOLY=.803+.1165*DLOG(NS)-.05123*DLOG(NS)**2-.01764*DLOG(NS)**3+
& .00503*DLOG(NS)**4

C ***** Compute uncorrected compressor adiabatic efficiency
ETACTS=((PS3/PT1)**((GAMMA-1.)/GAMMA)-1.)/((PS3/PT1)**2
& ((GAMMA-1.)/(ETAPOLY*GAMMA))-1.)
ETACTS=MIN(ETACTS,.95)
ETACTS=MAX(ETACTS,.6)
ETA1=ETACTS
C ***** Compute efficiency correction due to tip clearance based on GFSD
C ***** analyses of compressors for classified programs.

CR=MAX(.010,.003*RADHC)
SOH=CR/((RAD1C+RADNC)/2.-RADHC)
ETACTS=ETACTS*(1.-10.*SOH**1.5)
ETA2=ETACTS

```

```
C **** Correct Compressor Efficiency for Reynolds No. based on ASME
C **** Code PTC-10
```

```
RHO1=PRESS(1)*144./(RGAS*TEMP(1))
REC=RHO1*SPEED*(2.*RAD1C/12.)**2*PI/(30.*VISCOS(TEMP(1)))
IF (REC.LT.4.E6) THEN
ETACTS=1.+(ETACTS-1.)*(2.E6/REC)**.1
ELSE
ETACTS=1.+(ETACTS-1.)*(2.E6/4.E6)**.1
END IF
ETA3=ETACTS

ETACTS=MIN(ETACTS,.95)
ETACTS=MAX(ETACTS,.6)
IF (ISTGC.GE.16) THEN
ERRORC=
& 'AXIAL COMPRESSOR SOLUTION NOT POSSIBLE WITH LESS THAN 16 STAGES'
ETACOMP=.6
RETURN
END IF
ITER=ITER+1
IF (ITER.GE.50) THEN
ISTGC=ISTGC+1
GO TO 5
ELSE
IF (IPRINT.GE.5) THEN
WRITE (61,100) DHIDEAL,DELHACT,UHUBC,VX,VOACR1,RHO1
WRITE (61,110) A1,RADHC,RAD1C,TS1,C1,WT1
WRITE (61,120) MREL,TT2,ACR2,PSOPT2,PSOPT3,PS3
WRITE (61,130) QOPT2,PT2,PS2OLD,PS2,RHO2
WRITE (61,140) A2,RADNC,NS,ETAOLD,ETAPOLY,ETA1
WRITE (61,150) CR,SOH,ETA2,RHO1,REC,ETA3
END IF

IF(ABS(ETACTS-ETAOLD).GT.0.0001) GO TO 10
IF(ABS((PS3-PS3OLD)/PS3).GT.0.001) GO TO 10
END IF
ANSQC=A1*SPEED**2
```

```
C **** Compute approximate blade root stress levels in psi, Ti alloy
RHOTI=.164
SIGMA=PI/1800./386.*RHOTI*ANSQC
SIGPV=30000.
ANSQCL=SIGPV/SIGMA*ANSQC
IF (ANSQC.GT.ANSQCL) ERRORC=
& 'AXIAL COMPRESSOR OPERATING ABOVE A*N^2 LIMIT'
```

```
ETACOMP=ETACTS*((PT3/PT1)**((GAMMA-1.)/GAMMA)-1.)/((PS3/PT1)**2
& ((GAMMA-1.)/GAMMA)-1.)
IF (ERRORC.EQ.' ') THEN
ETACOMP=MAX(0.6,MIN(.95,ETACOMP))
```

```

UTIPC=2.*RAD1C/12.*PI*SPEED/60.
HMIN=RADNC-RADHC
IF (HMIN.LT.0.5) WARNINGC='AXIAL COMPRESSOR BLADE HEIGHT < .5 IN.'
RETURN
END IF
20 ISTGC=ISTGC+1
GO TO 5

100 FORMAT (' DHIDEAL,DELHACT,UHUBC,VX,VOACR1,RHO1'/1X,6E12.5)
110 FORMAT (' A1,RADHC,RAD1C,TS1,C1,WT1'/1X,6E12.5)
120 FORMAT (' MREL,TT2,ACR2,PSOPT2,PSOPT3,PS3'/1X,6E12.5)
130 FORMAT (' QOPT2,PT2,PS2OLD,PS2,RHO2'/1X,6E12.5)
140 FORMAT (' A2,RADNC,NS,ETAOLD,ETAPOLY,ETA1'/1X,6E12.5)
150 FORMAT (' CR,SOH,ETA2,RHO1,REC,ETA3'/1X,6E12.5)

END

SUBROUTINE AXTURB
C **** AXTURB is checked-out and final as of 10-06-92
$DEBUG
IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)

REAL JCON

CHARACTER*10 GAS(2),COMPTYPE,TURBTYPE,CLNTYPE,TVAR(18)
CHARACTER*20 GENTYPE,INTTYPE
CHARACTER*64 ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNINGT,WARNINGC,
& WARNINGR,WARNINGG

COMMON/OPTIM/VAR(18),TVAR,IPRINT

COMMON/DIAGNOS/ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNINGT,
& WARNINGC,WARNINGR,WARNINGG

COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDL
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
& RADNC,ANSQC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,
& RADHT,RAD1T,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT
COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,
& CPCLN, LIFETIME
COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTYPE
COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLOW,
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCTR,
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,
& DPOP16,DPOP17,NETSP,GROSSE,PFCYC,WINDAGE
COMMON/CONST/PI,RU,GO,JCON,RZERO

C **** SUBROUTINE INPUTS
C ****          PT1-INLET TOTAL PRESSURE, PSIA

```

C \*\*\*\* TT1-INLET TOTAL TEMPERATURE, R  
 C \*\*\*\* WDOT-MASS FLOWRATE, LBM/S  
 C \*\*\*\* N-TURBINE SPEED, RPM  
 C \*\*\*\* PS3-DIFFUSER EXIT STATIC PRESSURE, PSIA  
 C \*\*\*\* GAMMA-RATIO OF SPECIFIC HEATS  
 C \*\*\*\* RGAS-WORKING FLUID GAS CONSTANT, FT-LBF/LBM-R  
 C \*\*\*\* CP-WORKING FLUID SPECIFIC HEAT, BTU/LBM-R  
 C \*\*\*\* CR-CLEARANCE RATIO

C A1A	REAL	Required flow area at turbine inlet, sq in
C ACR1	REAL	Turbine inlet critical sonic velocity, ft/sec
C ACR2	REAL	Turbine rotor exit critical sonic velocity, ft/sec
C A2	REAL	Required flow area at last stage exit, sq in
C ANSQT	REAL	Flow area times speed squared parameter, sq in-rpm^2
C ANSQLT	REAL	Current AN^2 limit for turbine, sq in-rpm^2
C CR1	REAL	Turbine tip clearance, inches
C DELHACT	REAL	Actual delta enthalpy across turbine, BTU/lbm
C DHIDEAL	REAL	Isentropic delta enthalpy across turbine, BTU/lbm
C DELISTG	REAL	Indicated change in stage count used during sizing
C ETAPOLY	REAL	Polytropic efficiency from Balje reference, dmls
C ERRORT	CHAR*64	Turbine design error message
C ETA1	REAL	Turbine t-s ad. base efficiency, dmls
C ETA2	REAL	Turbine t-s ad. efficiency after clearance corr. dmls
C ETA3	REAL	Turbine t-s ad. efficiency after Re No. corr., dmls
C ETAOLS	REAL	Previous turbine t-s ad. overall efficiency, dmls
C ETATTS	REAL	Turbine t-s adiabatic efficiency [subject to corrections]
C ETATURB	REAL	Turbine t-t ad. efficiency overall, dmls
C H	REAL	Blade height, inches
C HMIN	REAL	Lower limit on blade height, inches
C ISTGT	REAL	Turbine stage count, dmls
C ITER	INTEG	Iteration counter
C LIFETIME	REAL	Design lifetime, years
C MUT	REAL	Turbine average fluid viscosity, lbm/ft-sec
C NS	REAL	Turbine specific speed [Balje form], dmls
C PT1	REAL	Turbine inlet total pressure, psia
C PS2	REAL	Turbine last stage exit static pressure, psia
C PS2OLD	REAL	Previous turbine last stage exit static pressure, psia
C PT2	REAL	Turbine last stage exit total pressure, psia
C PS3	REAL	Turbine diffuser discharge static pressure, psia
C PT3	REAL	Turbine diffuser discharge total pressure, psia
C PS2	REAL	Last stage exit static pressure, psia
C PT2	REAL	Turbine last stage discharge total pressure, psia
C PSOPT2	REAL	Static:total pressure ratio at last stage exit, dmls
C PSOPT3	REAL	Static:total pressure ratio at diffuser exit, dmls
C Q2	REAL	Turbine rotor discharge volume flow, cu ft/sec
C QOPT2	REAL	Turb. exit dynamic pressure:total pressure ratio, dmls
C RAD1T	REAL	First stage inlet flowpath outer radius, inches
C RADHT	REAL	Flowpath inner [hub] radius, inches
C RADNT	REAL	Last stage exit flowpath outer radius, inches
C REFFT	REAL	Turbine Reynolds Number, dmls

C	RGAS	REAL	Working fluid gas constant, ft-lbf/lbm-R
C	RHO1	REAL	First stage inlet fluid total density, lbm/cu ft
C	RHO1A	REAL	First stage inlet fluid density, lbm/cu ft
C	RHO2	REAL	Static density at last stageexit, lbm/cu ft
C	RHO3	REAL	Static density at turbine diffuser exit, lbm/cu ft
C	RHOTZM	REAL	Material density of TZM alloy, lbm/cu in
C	SO	REAL	Reference tip clearance [= 2% blade height], inches
C	SIGMA	REAL	Estimated blade root stree level, psi
C	SIGPV	REAL	Allowable blade root stree level, psi
C	SOSO	REAL	Clearance/reference clearance ratio, dmls
C	TT1	REAL	Turbine inlet total temperature, R
C	TT3	REAL	Turbine discharge total temperature, R
C	TS3	REAL	Turbine discharge static temperature, R
C	TT2	REAL	Turbine exit total temperature, R
C	TMATL	REAL	Temperature of hottest blade root material, deg R
C	UHUBT	REAL	Turbine rotor hub velocity, ft/sec
C	UTIPT	REAL	Rotor tips speed, ft/sec
C	VOACR1A	REAL	First stage rotor inlet absoluste Crit Vel Ratio, dmls
C	VU1A	REAL	Tangential velocity at first stage rotor inlet, ft/sec
C	VX	REAL	Meridinal velocity at first stage rotor inlet, ft/sec
C	WARNINGT	CHAR*64	Turbine design warning message
C	WDOT	REAL	Turbine stage fluid flowrate, lbm/sec

C \*\*\*\* ESTIMAT STAGE COUNT BASED ON HUB SPEED OF 1500 FT/S AND WORK FACTOR  
C \*\*\*\* OF 1.8 AND INITIAL GUESS OF .90 EFFICIENCY

```

RGAS=RU/MOLWT
DELISTG=0.
PT1=PRESS(10)
TT1=TEMP(10)
WDOT=FLOW(10)
PT3=PRESS(12)
PS3=PT3*.99
PT2=PT3*1.02
PSOPT2=.97
PS2=PT2*PSOPT2
ETATURB=.90
ETATTS=.90
ITER=0

```

```

10 CONTINUE
ERRORT=' '
WARNINGT=' '
DELHACT=ETATTS*TT1*CP*(1.-(PS3/PT1)**((GAMMA-1.)/GAMMA))
DHIDEAL=DELHACT/ETATTS
TT2=TT1-DHIDEAL*ETATTS/CP
TS2=TT2*PSOPT2**((GAMMA-1.)/GAMMA)
TT3=TT2
TS3=TT3*PSOPT3**((GAMMA-1.)/GAMMA)
RHO2=PS2*144./RGAS/TS2
Q2=ABS(WDOT/RHO2)

```

```

      IF (ITER.LT.35) ISTGT=INT(G0*JCON*DIDEAL/(SPEED*PI/30.*SQRT(Q2)/
& .55)**1.33333+DELISTG+1.)
      IF (ISTGT.LE.1) ISTGT=1
C **** CALCULATE 1st STAGE HUB STATOR EXIT MACH NUMBER (STA. 1a)

      ACR1=SQRT(2.*GAMMA/(GAMMA+1.)*G0*RGAS*TT1)
      UHUBT=SQRT(G0*JCON*DELHACT/(1.8*FLOAT(ISTGT)))

C **** STATOR EXIT TANGENTIAL VELOCITY

      VU1A=1.8*UHUBT

C **** ASSUME STATOR ANGLE OF 72 DEG

      VOACR1A=VU1A/SIN(72*PI/180)/ACR1

C **** FIRST STAGE GEOMETRY

      VX=VU1A/TAN(72*PI/180)
      RHO1A=PT1*144./RGAS/TT1*(1.-(GAMMA-1.)/(GAMMA+1.)*(VOACR1A)**2)
& **(1./(GAMMA-1.))
      A1A=WDOT/RHO1A/VX*144.
      RADHT=UHUBT/(SPEED*(PI/30.))*12.
      RAD1T=SQRT(A1A/PI+RADHT**2)

C **** TEST IF RADHT/RAD1T > .9, IF SO INCREASE ISTGT

      IF (RADHT/RAD1T.GT.0.9) THEN
      DELISTG=DELISTG+1
      ERROR='RADHT/RAD1T > 0.90'
      END IF

C **** COMPUTE EXIT TEMPERATURE

      TT2=TT1-DELHACT/CP
      ACR2=SQRT(2.*GAMMA/(GAMMA+1.)*G0*RGAS*TT2)
C **** COMPUTE EXIT STATIC:TOTAL PRESSURE RATIO
      TSOTT2=(1.-(GAMMA-1.)/(GAMMA+1.)*(VX/ACR2)**2)
      PSOPT2=TSOTT2**((GAMMA/(GAMMA-1.)))

C **** COMPUTE EXIT KINETIC ENERGY

      QOPT2=1.-PSOPT2

C **** COMPUTE ROTOR EXIT PRESSURE BASED ON .4 DIFFUSER CPR

      PS3=PT2*(PSOPT2+.4*QOPT2)
      PSOPT3=PS3/PT3

C **** EVALUATE LAST STAGE GEOMETRY
C **** EXIT VELOCITY

```

```

C      RHO2=144.*PT2/RGAS/TT2*(1.-(GAMMA-1.)/(GAMMA+1.)*(VX/ACR2)**2)
C      & **(1./(GAMMA-1.))
C      RHO2=144.*PS2/RGAS/TS2

C ***** EXIT AREA
A2=WDOT/RHO2/VX*144.

C ***** TIP RADIUS
RADNT=SQRT(A2/PI+RADHT**2)

C ***** IF RADHT/RADNT < .6, DECREASE STAGE COUNT
IF ((RADHT/RADNT).LT.0.6) THEN
DELISTG=DELISTG-1.
ERRQRT='RADHT/RADNT < 0.60'
END IF

C ***** EFFICIENCY ESTIMATES - BALJE'S Ns,t CORRELATION
PT2=PT3+.156*PT2*QOPT2
PS2OLD=PS2

C ***** COMPUTE ROTOR EXIT STATIC PRESSURE
PS2=PT2*PSOPT2

TS3=TT2*(PS3/PT3)**((GAMMA-1.)/GAMMA)
RHO3=PS3*144./RGAS/TS3
NS=SPEED*PI/30.*SQRT(WDOT/RHO3)/(GO*JCON*DIDEAL/DFLOAT(ISTGT))
& **.75
ETA0LD=ETATTs

C ***** Compute axial turbine uncorrected polytropic efficiency
C ***** Curve fit of data taken from:
C           Balje, O. J., "Turbomachines: A Guide to Design, Selection,
C           and Theory", John Wiley and Sons, Inc., New York, 1981.

ETAPOLY=.86869509-.055979102*DLOG(NS)-.04902661*DLOG(NS)**2+
& .0022114298*DLOG(NS)**3
ETATTs=(1.-(PS3/PT1)**(ETAPOLY*((GAMMA-1.)/GAMMA)))/(1.-
& (PS3/PT1)**((GAMMA-1.)/GAMMA))
ETA1=ETATTs

C ***** CORRECTION FOR TIP CLEARANCE
CR1=MAX(.003*RADHT,.01)
H=RAD1T-RADHT
S0=.02*H
SOS0=CR1/S0

```

```

IF (SOS0.GT.1.) THEN
ETATTS=ETATTS*(1.0108-.01*SOS0-8.33E-4*SOS0**2)
ELSE
ETATTS=ETATTS
END IF
ETA2=ETATTS

```

```

C **** Correct Turbine Efficiency for Reynolds No. based on ASME
C **** Code PTC-10

```

```

MUT=AVVISC(TEMP(10),TEMP(12))
RH01=PT1*144./(RGAS*TEMP(10))
REFFT=RH01*SPEED*(2.*RAD1T/12.)**2*PI/(30.*MUT)
IF (REFFT.LT.4.E6) THEN
ETATTS=1.+(ETATTS-1.)*(2.E6/REFFT)**.2
ELSE
ETATTS=1.+(ETATTS-1.)*(2.E6/4.E6)**.2
END IF
ETA3=ETATTS

```

```

IF (IPRINT.GE.4) THEN
WRITE (61,100) DELHACT,DHIDEAL,TT2,TT3,RHO2,Q2
WRITE (61,110) ACR1,UHUBT,VU1A,VOACR1A,VX,RHO1A
WRITE (61,120) A1A,RADHT,RAD1T,TT2,ACR2,PSOPT2
WRITE (61,130) QOPT2,PS3,PS2OLD,PS2,RHO2,A2
WRITE (61,140) RADNT,PT3,TS3,RHO3,NS,ETAPOLY
WRITE (61,150) ETA1,CR1,H,S0,SOS0,ETA2
WRITE (61,160) PT2,ETA3,ISTGT
END IF

```

```

ITER=ITER+1
IF (ITER.GT.50) THEN
ERRORT='FAILED TO CONVERGE IN 50 ITERATIONS IN SUBROUTINE AXTURB'
ETATURB=.6
RETURN
END IF

```

```

IF (ABS((PS2-PS2OLD)/PS2).GT.0.0001) GO TO 10
IF (ABS(ETATTS-ETAOLD).GT.0.00001) GO TO 10
ANSQT=A2*SPEED**2

```

```

C **** Compute approximate blade root stress levels in psi, TZM alloy
RHOTZM=.37
SIGMA=PI/1800./386.*RHOTZM*ANSQT
TMATL=TEMP(10)

```

```

C      WRITE (*,234) SIGMA,TMATL,LIFETIME
C 234 FORMAT (' SIGMA,TMATL,LIFETIME ',3E12.5)

```

```

CALL STRNTH (TMATL,LIFETIME,SIGPV)
ANSQTL=SIGPV/SIGMA*ANSQT

```

```

C      WRITE (*,235) SIGPV,ANSQTL
C 235 FORMAT (' SIGPV,ANSQTL   '3E12.5)

      IF (ANSQT.GT.ANSQTL) ERRORT=
      & 'AXIAL TURBINE OPERATING ABOVE A*N^2 LIMIT; RED. TIT OR INC. MW'

      RHO1T=144*PRESS(10)/(RGAS*TEMP(10))
      RHO2T=144*PRESS(12)/(RGAS*TEMP(12))
      TURBSS=SPEED*(WDOT**2/RHO1T/RHO2T)**.25/(JCON*CP*(TEMP(10)-
      & TEMP(12)))**.75
      UTIPT=2.*PI*RADNT/12.*SPEED/60.

C ***** Convert from t-s to t-t overall efficiency

      ETATURB=ETATTS*(1.-(PS3/PT1)**((GAMMA-1)/GAMMA))/(
      & (1.-(PT3/PT1)**((GAMMA-1)/GAMMA)))
      ETATURB=MAX(0.6,MIN(.98,ETATURB))

      ETATURB=MAX(0.6,MIN(.95,ETATURB))
      HMIN=RAD1T-RADHT
      IF (HMIN.LT.0.5) WARNINGC='AXIAL TURBINE BLADE HEIGHT < .5 IN.'

      RETURN
100 FORMAT (' DELHACT,DHIDEAL,TT2,TT3,RHO2,Q2 ',1X,6E12.5)
110 FORMAT (' ACR1,UHUBT,VU1A,VOACR1A,VX,RHO1A',1X,6E12.5)
120 FORMAT (' A1A,RADHT,RAD1T,TT2,ACR2,PSOPT2',1X,6E12.5)
130 FORMAT (' QOPT2,PS3,PS2OLD,PS2,RHO2,A2',1X,6E12.5)
140 FORMAT (' RADNT,PT3,TS3,RHO3,NS,ETAPOLY',1X,6E12.5)
150 FORMAT (' ETA1,CR1,H,SO,SOSO,ETA2',1X,6E12.5)
160 FORMAT (' PT2,ETA3,ISTGT',1X,2E12.5,I5)

      END

```

**BRAY3.FOR**

```
$DEBUG
$NOTRUNCATE
      SUBROUTINE RECSIZE
C **** RECSIZE is checked-out and final on October 6, 1992

      IMPLICIT REAL (A-H)
      IMPLICIT INTEGER (I-J)
      IMPLICIT REAL (K-Z)
      REAL JL,JH,JCON

      CHARACTER*10 GAS(2),TVAR(18)

      CHARACTER*64 ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,WARNIGC,
& & WARNINGR,WARNIGG

      COMMON/DIAGNOS/ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNIGT,
& & WARNINGC,WARNINGR,WARNIGG

      COMMON/OPTIM/ VAR(18),TVAR,IPRINT
      COMMON/OUTP/WGTTOT,WGTTAC,RECLC,RECLOA,RECH,RECW
      COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
      COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLO,
& & PRC,EFFC,EFFT,E,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCCTR,
& & DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,DPOP16,
& & DPOP17,NETSP,GROSSEP,EFFCYCLE,WINDAGE
      COMMON/RECUP/ DPDES,UAML,UAMH,DPML,DPMH,
# PREXP,NFINL,TFINL,SPACEL,LEQL,ROFINL,KFINL,NFINH,TFINH,SPACEH,
# LEQH,ROFINH,KFINH,TPLATE,ROPLAT,TBRAZE,ROBRAZ,
# SIGMAL,ALPHAL,AFINL,APLATL,HDL,SIGMAH,ALPHAH,AFINH,APLATL,HDH,
# PITCH,WGTREC
      COMMON/MASSES/MASSCMP,MASSTRB,MWHLC,MWHLT,MSFTC,MSFTT,MCASEC,
& & MCASET,MDIAC,MDIAT

      COMMON/CONST/PI,RU,GO,JCON,RZERO
```

```
C -----
C -----
C   COUNTERFLOW PLATE-FIN RECUPERATOR ANALYSIS
C -----
C   -----NOMENCLATURE-----
C   -----OPTIONS AND COUNTERS-----
C
C   FOR THESE PRINT OPTIONS
C   0      = NO
C   1      = YES
C
C   IPRINT = PRINT OPTION
C
C   AFIN   = FIN AREA:VOLUME RATIO (SQ IN/CU IN)
C   AFRONT = CORE FRONTAL AREA (SQ IN)
C   ALPHA  = HEAT TRANSFER AREA:VOLUME RATIO (SQ IN/CU IN)
C   APLATE = PLATE OR TUBE AREA:VOLUME RATIO (SQ IN/CU IN)
```

C COND = HEAT TRANSFER CONDUCTANCE (BTU/SEC-R)  
C CP = FLUID SPECIFIC HEAT (BTU/LBM-R)  
C DP = PRESSURE LOSS (PERCENT OR PSID)  
C DPCORE = CORE PRESSURE DROP (PSID)  
C E = RECUPERATOR TEMPERATURE EFFECTIVENESS (DMLS)  
C ETAFIN = FIN HEAT TRANSFER EFFECTIVENESS (DMLS)  
C F = FRICTION FACTOR  
C FLOW = MASS FLOW RATE (LBM/SEC)  
C H = SURFACE HEAT TRANSFER FILM CONDUCTANCE (BTU/SQ IN-SEC-R)  
C HCORE = CORE HEIGHT (INCHES)  
C HD = PASSAGE HYDRAULIC DIAMETER (INCHES)  
C J = COLBURN HEAT TRANSFER "J-FACTOR" (DMLS)  
C K = MATERIAL THERMAL CONDUCTIVITY (BTU/SEC-IN-R)  
C LEQ = EQUIVALENT FIN CONDUCTION LENGTH (INCHES)  
C LCORE = CORE LENGTH (INCHES)  
C MU = FLUID VISCOSITY (LB/FT-SEC)  
C MOLWT = FLUID MOLECULAR WEIGHT (LB/LB-MOLE)  
C NFIN = NUMBER OF FINS PER INCH (DMLS)  
C P = PRESSURE (PSIA)  
C PITCH = THICKNESS OF REPEATING CORE SANDWICH (INCHES)  
C PR = FLUID PRANDTL NUMBER (DMLS)  
C RU = UNIVERSAL GAS CONSTANT (FT-LBF/R-LB MOLE)  
C REY = REYNOLDS NUMBER (DMLS)  
C SIGMA = FREE-FLOW AREA:FRONTAL AREA RATIO (DMLS)  
C SPACE = PLATE SPACING (INCHES)  
C RHOFIN = FIN MATERIAL DENSITY (LBM/CU FT)  
C ROPLAT = SPLITTER PLATE MATERIAL DENSITY (LBM/CU FT)  
C T = TEMPERATURE (DEG R)  
C TBRAZ = CORE BRAZE MATERIAL THICKNESS (INCHES)  
C TFIN = FIN MATERIAL THICKNESS (INCHES)  
C TPLATE = SEPARATOR PLATE MATERIAL THICKNESS (INCHES)  
C UA = HEAT TRANSFER OVERALL CONDUCTANCE (BTU/S-R)  
C WCORE = CORE WIDTH (INCHES)

C -----  
C SUFFIX DEFINITIONS

C xH = High-pressure flowpath  
C xL = Low-pressure flowpath

C xIH = Inlet to high-pressure passage  
C xDH = Discharge from high-pressure passage  
C xIL = Inlet to low-pressure passage  
C xDL = Discharge from low-pressure passage

C \*\*\*\* Assumptions -

- C 1) 30% of the pressure drop allocation is consumed in the entry/exit  
C transitions. Therefore the core is designed at 70% of the allocated  
C pressure loss.  
C 2) The recuperator entry/exit transition sections, end-sections and  
C headers add 50% to the core mass.

C \*\*\*\* Check the input data file for the presence of a recuperator. If  
C \*\*\*\* Er = 0 an unrecuperated cycle design is assumed.

```
IF (E.LE.0.) THEN
WTREC=0.
DPOP7=0.
DPOP14=0.
RETURN
END IF
```

C \*\*\*\* Initialize and equate recuperator design variables

```
ERRORR=' '
WARNINGR=' '
DPFRACT=.7
LCORE=20.
REYH=300.
TIH=TEMP(6)
PIH=PRESS(6)
TIL=TEMP(13)
PIL=PRESS(13)
FLOWL=FLOW(13)
FLOWH=FLOW(6)
TDH=TIH+E*(TIL-TIH)
TDL=TIL-E*FLOWH/FLOWL*(TIL-TIH)
```

C \*\*\*\* Compute log-mean temperature difference and required conductance.

```
LMDT=((TIL-TDH)+(TDL-TIH))/2.
UAREQD=FLOWH*CP*(TDH-TIH)/LMDT
```

C \*\*\*\* Compute values for design constant groupings.

```
CON3L=2.*LEQL**2/(KFINL/3600./12.*TFINL)
```

```
CON3H=2.*LEQH**2/(KFINH/3600./12.*TFINH)
```

```
CON10=2.*TPLATE/PITCH
CON11L=NFINL*TFINL*(SPACEL+1./NFINL-TFINL)/PITCH
CON11H=NFINH*TFINH*(SPACEH+1./NFINH-TFINH)/PITCH
```

C \*\*\*\* Define viscosity and Prandtl No. for high-pressure flow passage.

```
TDH=TIH+E*(TIL-TIH)
DELHH=CP*(TDH-TIH)
MUH=AVVISC(TIH,TDH)
PRH=CP*MUH/AVCOND(TIH,TDH)
```

C \*\*\*\* Define viscosity and Prandtl No. for low-pressure flow passage.

```
DELHL=-DELHH*FLOWH/FLOWL  
TDL=TIL+DELHL/CP  
DELHL=CP*(TDL-TIL)  
MUL=AVVISC(TIL,TDL)  
PRL=CP*MUL/AVCOND(TIL,TDL)
```

C \*\*\*\* Define thermal-capacity-rate-ratio [TCRAT] and low-pressure side Er.

```
TCRAT=FLOWH/FLOWL  
IF (ABS(TCRAT-1.).LT.1.E-4) TCRAT=1  
EL=E*TCRAT  
  
PML=PIL  
PMH=PIH  
PDL=PIL  
PDH=PIH  
TML=(TIL+TDL)/2.  
TMH=(TIH+TDH)/2.
```

C \*\*\*\* Compute values for design constant groupings.

```
CON1L=HDL/(MUL*SIGMAL)*12.  
CON1H=HDH/(MUH*SIGMAH)*12.  
CON2L=(CP*MUL)/(PRL**PREXP*HDL)/12.*(0.7**PREXP)/(0.7**.66667)  
CON2H=(CP*MUH)/(PRH**PREXP*HDH)/12.*(0.7**PREXP)/(0.7**.66667)
```

```
ITER=0  
10 FRONT=FLOWH/REYH*CON1H  
REYL=FLOWL/FRONT*CON1L
```

C \*\*\*\* Compute friction factor and Colburn j-factor for both flow passages.

```
FH=10**(2.307556-1.904247*ALOG10(REYH)+.2408641*ALOG10(REYH)**2)  
FL=10**(2.307556-1.904247*ALOG10(REYL)+.2408641*ALOG10(REYL)**2)  
JH=10**(1.196432-1.524831*ALOG10(REYH)+.1688725*ALOG10(REYH)**2)  
JL=10**(1.196432-1.524831*ALOG10(REYL)+.1688725*ALOG10(REYL)**2)  
HL=CON2L*JL*REYL  
HH=CON2H*JH*REYH
```

C \*\*\*\* Compute the double-sided fin effectiveness.

```
BRACL=SQRT(HL*CON3L)  
BRACH=SQRT(HH*CON3H)  
ETAFNL=TANH(BRACL)/BRACL  
ETAFNH=TANH(BRACH)/BRACH
```

C \*\*\*\* Compute the overall surface heat transfer effectiveness.

```
ETAL=1.-AFINL/ALPHAL*(1.-ETAFNL)  
ETAH=1.-AFINH/ALPHAH*(1.-ETAFNH)
```

```

C **** Compute Eta*HAs here w/o axial conduction influence.

    ETAHAL=ETAL*HL*(AFINL+APLATL)*FRONT*LCore
    ETAAH=ETAH*HH*(AFINH+APLATH)*FRONT*LCore

C **** Compute the overall heat transfer conductance for this iteration.

    UACALC=1./((1.+UAML)/ETAHAL+(1.+UAMH)/ETAAH)
    LCore=LCore*UAREQD/UACALC

C **** For task Order 18 studies, no end section is computed; LENGTH=LCore

    LENGTH=LCore
    CON6L=((MUL/12.)*REYL/(HDL*PML))**2*(RU/MOLWT*TIL/(2.*GO))
    CON6H=((MUH/12.)*REYH/(HDH*PMH))**2*(RU/MOLWT*TIH/(2.*GO))
    DPCORL=CON6L*(4.*FL*LCore/HDL)*(TML/TIL)*(1+DPML)
    DPCORH=CON6H*(4.*FH*LCore/HDH)*(TMH/TH)*(1+DPMH)
    DPOPTL=1.-(1.-DPCORL)*(1.-DPCORH)
    PDL=PIL*(1.-DPCORL)
    PDH=PIH*(1.-DPCORH)
    PML=(PIL+PDL)/2.
    PMH=(PIH+PDH)/2.

C **** Estimate the value of high-press side Reynold's No. to achieve
C **** convergence on the desired fractional pressure loss.

    REYH=REYH*SQRT(DPFRACT*DPDES/DPOPTL)

    ITER=ITER+1
    IF (ITER.GE.100) ERRORR=' ITERATION LIMIT EXCEEDED IN RECUPERATOR
#ROUTINE'

C **** Check for the presence of an error message.

    IF (ERRORR.NE.' ') GO TO 30

C **** Check for convergence of calculated and desired pressure loss.

    IF (ABS((DPFRACT*DPDES-DPOPTL)/(DPFRACT*DPDES)).GT..001) GO TO 10

30  WCORE=SQRT(FRONT)
    HCORE=SQRT(FRONT)

C ***** Basic core mass only ***** June 12, 1992

C **** Compute the weight components of the core design defined above.

    WPLATE=CON10*HCORE*WCORE*LCore*ROPLAT
    WBRAZE=2.*(TBRAZE/TPLATE)*(ROBRAZ/ROPLAT)*WPLATE
    WFINL=FRONT*CON11L*LCore*ROFINL

```

```

WFINH=FRONT*CON11H*LCORE*ROFINH
WBARS=RZERO
WWRAP=RZERO
WHDRL=RZERO
WHDRH=RZERO
VOLUME=RZERO
WFLOWL=RZERO
WFLOWH=RZERO
WGTREC=(WPLATE+WFINL+WFINH+WBARS+WWRAP+WHDRL+WHDRH+WBRAZE+WFLOWL+
#WFLOWH)
C **** Adding a 50% allowance for end-sections/ wrap-up
WGTREC=WGTREC*1.5
CONDL=ETAL*HL*FRONT*LCORE*ALPHAL
CONDH=ETAH*HH*FRONT*LCORE*ALPHAH

C **** Print design summary to File:PROG$OUT is IPRINT.GT.0
IF (IPRINT.GE.1) THEN
C ** convert to SI for printout
XTIL=TIL/1.8
XTDL=TDL/1.8
XTIH=TIH/1.8
XTDH=TDH/1.8

XFLOWL=FLOWL*0.45359
XPIL=PIL*6.8948
XFLOWH=FLOWH*0.45359
XPIH=PIH*6.8948
XCONDL=CONDL*1054.4*1.8
XMUL=MUL*1488.
XCP=CP
XCONDH=CONDH*1054.4*1.8
XMUH=MUH*1488.

XWHDRL=WHDRL*.45359
XWFINL=WFINL*.45359
XWFLOWL=WFLOWL*.45359
XWHDRL=WHDRL*.45359
XWFINH=WFINH*.45359
XWFLOWH=WFLOWH*.45359

XWGTREC=WGTREC*.45359
XWPLATE=WPLATE*.45359
XWBARS=WBARS*.45359
XWWRAP=WWRAP*.45359
XWBRAZE=WBRAZE*.45359

XLCORE=LCORE*2.54
XCORE=WCORE*2.54
XHCORE=HCORE*2.54

```

```

XLENGTH=LENGTH*2.54
XVOLUME=VOLUME*(12**3.)*(2.54**3.)

      WRITE (61,100)
      WRITE (61,120)
      WRITE (61,110)
      WRITE (61,120)
      WRITE (61,180) GAS(1),GAS(2),GAS(1),GAS(2)
      WRITE (61,120)
      WRITE (61,190)
      WRITE (61,130) EL,XTIL,XTDL,E,XTIH,XTDH
      WRITE (61,200)
      WRITE (61,210)
      WRITE (61,130) XFLOWL,XPIL,DPCORL,XFLOWH,XPIH,DPCORH
      WRITE (61,220)
      WRITE (61,130) ETAFNL,REYL,PRL,ETAFNH,REYH,PRH
      WRITE (61,230)
      WRITE (61,130) XCONDL,XMUL,XCP,XCONDH,XMUH,XCP
      WRITE (61,240)
      WRITE (61,130) XWHDRL,XWFINL,XWFLOWL,XWDRH,XWFINH,XWFLOWH
      WRITE (61,100)
      WRITE (61,160)
      WRITE (61,250)
      WRITE (61,160)
      WRITE (61,140) XWGTREC,XWPLATE,XWBARS,XWWRAP,XWBRAZE,RZERO
      WRITE (61,150)
      WRITE (61,260)
      WRITE (61,160)
      WRITE (61,140) XLCORE,XWCORE,XHCORE,XLENGTH,XVOLUME,DPDES
      WRITE (61,150)
      WRITE (61,100)
END IF
RETURN

100 FORMAT (1X,127(1H*))
115 FORMAT (1X,'FAILED TO CIRCLE IN 50 PASSES, TOO BAD. ERRASP=',F8.4)
110 FORMAT (1X,1H*,22X,17HLOW PRESSURE SIDE,23X,1H*,22X,18HHIGH PRESSU
#RE SIDE,22X,1H*)
120 FORMAT (1X,1H*,62X,1H*,62X,1H*)
130 FORMAT (1X,1H*,62X,1H*,62X,1H*/,1X,1H*,2(5X,E12.5,2(8X,E12.5),
#5X,1H*),/,1X,1H*,62X,1H*,62X,1H*/,1X,1H*,62X,1H*,62X,1H*)
140 FORMAT (1X,1H*,5X,E12.5,2(8X,E12.5),11X,E12.5,2(8X,E12.5),5X,1H*)
150 FORMAT (1X,1H*,125X,1H*/,1X,1H*,125X,1H*)
155 FORMAT (5X,'RECUPERATOR TOTAL COST,$',F12.0)
160 FORMAT (1X,1H*,125X,1H*)
170 FORMAT (1X,51X,28HPLATE FIN COUNTERFLOW DESIGN,/ )
180 FORMAT (1X,1H*,23X,7HGAS IS ,2A10,12X,1H*,23X,7HGAS IS ,2A10,12X,1
#H*)
190 FORMAT (1X,1H*,4X,13HEFFECTIVENESS,4X,17HINLET TEMPERATURE,5X,16HE
#XIT TEMPERATURE,3X,1H*,4X,13HEFFECTIVENESS,4X,17HINLET TEMPERATURE
#,5X,16HEXIT TEMPERATURE,3X,1H*/,1X,1H*,7X,7H(-----),13X,8H(DEG. K
#),12X,8H(DEG. K),7X,1H*,7X,7H(-----),13X,8H(DEG. K),12X,8H(DEG. K)

```

```

#,7X,1H*)
200 FORMAT (1X,1H*,7X,9HFLOW RATE,8X,14HINLET PRESSURE,7X,13HPRESSURE
#DROP,4X,1H*,7X,9HFLOW RATE,8X,14HINLET PRESSURE,7X,13HPRESSURE DRO
#P,4X,1H*)
210 FORMAT (1X,1H*,7X,8H(KG/SEC),13X,6H(KPA),13X,9H (DP/P) ,6X,1H*,7
#X,8H(KG/SEC),13X,6H(KPA),13X,9H (DP/P) ,6X,1H*)
220 FORMAT (1X,1H*,2X,17HFIN EFFECTIVENESS,6X,12HREYNOLDS NO.,8X,11HPR
#ANDTL NO.,6X,1H*,2X,17HFIN EFFECTIVENESS,6X,12HREYNOLDS NO.,8X,11H
#PRANDTL NO.,6X,1H*/,,1X,1H*,7X,7H(-----),13X,7H(-----),13X,7H(-----
#-),8X,1H*,7X,7H(-----),13X,7H(-----),13X,7H(-----),8X,1H*)
230 FORMAT (1X,1H*,3X,16HH.T. CONDUCTANCE,7X,9HVISCOSITY,9X,13HSPECIFI
#C HEAT,5X,1H*,3X,16HH.T. CONDUCTANCE,7X,9HVISCOSITY,9X,13HSPECIFIC
# HEAT,5X,1H*/,,1X,1H*,7X,10H(W/ DEG.K) ,11X,4H(CP),12
#X,14H(CAL/GM-DEG.K),4X,1H*,7X,10H(W/ DEG.K) ,11X,4H(CP),12X,
#14H(CAL/GM-DEG.K),4X,1H*)
240 FORMAT (1X,1H*,5X,13HHEADER WEIGHT,9X,10HFIN WEIGHT,9X,12HFLUID WE
#IGHT,4X,1H*,5X,13HHEADER WEIGHT,9X,10HFIN WEIGHT,9X,12HFLUID WEIGH
#T,4X,1H*/,,1X,1H*,9X,4H(KG),17X,4H(KG),16X,4H(KG),8X,1H*,9X,4H(KG)
#,17X,4H(KG),16X,4H(KG),8X,1H*)
250 FORMAT (1X,1H*,5X,12HTOTAL WEIGHT,8X,12HPLATE WEIGHT,6X,17HSPACER
#BAR WEIGHT,7X,14HWRAP UP WEIGHT,7X,12HBRAZE WEIGHT,9X,12H
#,4X,1H*/,,1X,1H*,9X,4H(KG),16X,4H(KG),16X,4H(KG),19X,4H(KG),16X
#,4H(KG),16X,7H(-----),6X,1H*)
260 FORMAT (1X,1H*,5X,11HCORE LENGTH,10X,10HCORE WIDTH,10X,11HCORE HEI
#GHT,10X,14HOVERALL LENGTH,10X,6HVOLUME,7X,19HTOTAL PRESSURE DROP,2
#X,1H*/,,1X,1H*,7X,4H(CM),2(17X,4H(CM)),16X,4H(CM),18X,
#7H(CU.CM),13X,9H(DP/P) ,5X,1H*)
280 FORMAT (1X,'CASE NO =',I4/)
300 FORMAT (50X,8A10)
310 FORMAT (1H1,48HTDH,DELHH,CPH,MUH,PRH = ,6E
#12.5)
320 FORMAT (1X,48HTDL,DELHL,CPL,MUL,PRL = ,6E1
#2.5)
330 FORMAT (1X,48HTCRAT,CON1L,CON1H,CON2L,CON2H = ,6E1
#2.5)
340 FORMAT (1X,48HCON3L,CON3H,COSTL,COSTH,SINTL,SINTH = ,6E1
#2.5)
350 FORMAT (1X,48HTANTL,TANTH,CON4L,CON4H,CON5 = ,6E1
#2.5)
360 FORMAT (1X,48HCON7L,CON7H,CON8L,CON8H,FRONT,REYL = ,6E1
#2.5)
370 FORMAT (1X,48HFL,JL,FH,JH = ,6E1
#2.5)
390 FORMAT (1X,48HHL,HH,BRACL,BRACH,ETAL,ETAH = ,6E1
#2.5)
400 FORMAT (1X,48HBRAC,LCORE,LHDRL,LHDRH,REHTRL,REHDRH = ,6E1
#2.5)
410 FORMAT (1X,48HFHTRL,FHDRH,CON6L,CON6H,DPCORL,DPCORH = ,6E1
#2.5)
420 FORMAT (1X,48HDPCTRL,DPCTRHL,DPHTRL,DPHDRH,DPEXPL,DPEXPH = ,6E1
#2.5)
430 FORMAT (1X,48HDPTRNL,DPTRNH,DPOPL,DPOPH,DPOPTL = ,6E1

```

```

#2.5)
440 FORMAT (1X,48HPDL,PDH,PML,PMH          = ,6E1
#2.5)
450 FORMAT (1X,48HPITCH,CON10,CON11L,CON11H,CON12L,CON12H      = ,6E1
#2.5)
460 FORMAT (1X,48HDTL0,DTLL,D2TL0,D2TLL,TDL,EXPL3      = ,6E1
#2.5)
470 FORMAT (1X,51HD13,D14,D31,D32,D33,D42
#6E12.5/52H D43,R1,R2,DETA,DETB,DETC      = ,
#2.5/52H DETD,DETE,A,B,C,D      =,6E1
#)
480 FORMAT (1X,44HC1,C2,C3,PP,QQ,RR      =,6E12.5)
490 FORMAT (1X,44HAA,BB,COSPHI,PHI      =,6E12.5)
500 FORMAT (1X,44H(M(I),I=1,3)      =,6E12.5)
510 FORMAT (39X,7HEFFECT.,4X,8HTT INLET,4X,9HTT DISCH.,5X,4HFLOW,6X,8H
#PT INLET,3X,11HPRANDTL NO.,6X,3HGAS,/39X,7H(-----),2(4X,8H(DEG. R)
#),4X,8H(LB/SEC),5X,6H(PSIA),6X,(7H(-----),5X)/)
520 FORMAT (4X,32HLOW PRESSURE SIDE -----,F9.4,3X,F9.2,3X,F10
#.2,2X,F10.3,2X,F9.2,3X,F9.5,6X,2A10)
530 FORMAT (4X,32HHIGH PRESSURE SIDE -----,F9.4,3X,F9.2,3X,F10
#.2,2X,F10.3,2X,F9.2,3X,F9.5,6X,2A10)
540 FORMAT (14X,9HL.P. DP/P,3X,9HH.P. DP/P,2X,95HTOTAL DP/P
# /WEIGHT VOLUME CORE LENGTH HEIGHT WIDTH TOTA
#L LENGTH,/,15X,4(7H(-----),5X),1X,4H(LB),6X,8H(CU.FT.),4(4X,8H(INC
#HES))/)
550 FORMAT (12X,3(F9.5,3X),F9.3,F13.3,      E11.4,1X,4(F10.3,2X))
560 FORMAT (/,1X,130(1H*))
570 FORMAT (1X,6E12.5)
END

```

#### SUBROUTINE DUCTING

C \*\*\*\* DUCTING is checked-out and final on October 6, 1992

```

IMPLICIT REAL (A-H, K-Y)
IMPLICIT INTEGER (I-J)
CHARACTER*60 ERR1, ERR2, ERR3, ERR4, ERR5, ERR6, ERR7, ERR8,
& ERR9, ERR10, ERR11, ERR12
COMMON /DUCTI1/ IDUCT, PIN, POUT, MDOT, TIN, MW
COMMON /DUCTI2/ EL1(6), EL2(6), TWL(6), DNWL(6), NMFI(6),
& TMFI(6), DMFI(6), ETA(6)
COMMON /DUCTOUT/ WGTDUCT(6), DIAM(6), LGTH(6), VGAS(6)
COMMON /DUCTERR/ ERR1, ERR2, ERR3, ERR4, ERR5, ERR6, ERR7, ERR8,
& ERR9, ERR10, ERR11, ERR12

```

```

C** INPUT DATA FROM THE CYCLE CODE
C** PIN - inlet pressure to the duct section, psia
C** POUT - outlet pressure from the duct section, psia
C** TIN - temperature at the inlet to the duct section, R
C** MDOT - mass flow rate in the duct section, lb/s
C** MW - molecular weight of the gas, 1bm/1bmole

```

C\*\* DATA FROM THE INPUT DATA FILE

C\*\* IDUCT - duct section identifier  
C\*\* iduct=1 - duct section between statepoints 5 and 6  
C\*\* iduct=2 - duct section between statepoints 7 and 8  
C\*\* iduct=3 - duct section between statepoints 9 and 10  
C\*\* iduct=4 - duct section between statepoints 12 and 13  
C\*\* iduct=5 - duct section between statepoints 14 and 15  
C\*\* iduct=6 - duct section between statepoints 16 and 17

C\*\* EL1(iduct) - L/Ds of straight duct in duct i  
C\*\* EL2(iduct) - L/D equivalent for elbows, etc, in duct i  
C\*\* TWL(iduct) - wall thickness for duct i  
C\*\* DNWL(iduct) - density of the wall material for duct i  
C\*\* NMFI(iduct) - number of layers of MFI on the duct  
C\*\* TMFI(iduct) - thickness of individual MFI foil, in  
C\*\* DMFI(iduct) - bulk density of the MFI system, lb/ft<sup>3</sup>  
C\*\* ETA(iduct) - surface roughness of the duct i

C\*\* He-Xe GAS PROPERTIES  
C\*\* CP - gas heat capacity, Btu/lb-F  
C\*\* DN - gas density, lbm/cu ft  
C\*\* PN - Prandtl No.  
C\*\* KG - thermal conductivity, Btu/hr-ft-F  
C\*\* VS - viscosity, lbm/hr-ft

C\*\* COMPUTED PARAMETERS TO BE PASSED BACK TO MANE  
C\*\* WGTDUCT(iduct) - total weight of duct segment, lbm  
C\*\* DIAM(iduct) - inside diameter of duct segment iduct, in  
C\*\* LGTH(iduct) - length of duct run iduct, in  
C\*\* VGAS(iduct) - velocity of the gas in duct i, ft/s

PI=3.14159265359  
GC=32.1739

PP=PIN  
TT=TIN - 460.  
DP=PIN - POUT

CALL HEXEN(MW,PP,TT,CP,DN,KG,VS)

C\*\* INITIAL GUESS OF DUCT GAS VELOCITY, FT/SEC  
VELGAS = 75.0

C\*\* INITIALIZE ITERATIONS  
NITER = 0.0  
DO 100 I = 1,1000

C\*\* CALCULATED DIAMETER (in) FROM VELOCITY

```

DID = (DSQRT((4.0*MDOT)/(DN*VELGAS*PI)))*12.0

C** REYNOLDS NUMBER
    RE = VELGAS*DN*(DID/12.0)/(VS/3600.0)

C** PRESSURE COEFFICIENT
    PC = ((MDOT/((PI*(DID)**2.0)/4.0))**2.0)*144.0/(2.0*GC*DN)

C** FF, FRICTION FACTOR
C** ETA, SURFACE ROUGHNESS, in
    FF = 0.25/(ALOG10(ETA(IDUCT)/(3.7*DID) + 5.74/RE**0.9))**2.0
    IF (RE.LE.2000.0) FF = 64.0/RE
    IF (RE.GT.2000.0 .AND. RE.LT.4000.0)
    & FF = 0.032*(1.0 - (RE-2000.0)/2000.0) + FF*(RE-2000.0)/2000.0

C** DP, CALCULATED PIPE PRESSURE DROP, 1bm/sq in
C** EL1, L/D FOR STRAIGHT PIPE
C** EL2, L/D FOR LOSSES
    DPC = FF*(EL1(IDUCT)+EL2(IDUCT))*PC

    ERROR = (DPC-DP)/DP
C** CHECK PRESSURE DROP
    IF (ABS(ERROR).LT.0.002) GO TO 200

    NITER = NITER + 1.0

    VELGAS = VELGAS*(1 - (0.1*ERROR))
C      WRITE (*,99) NITER, VELGAS
C 99  FORMAT (' ITERATIONS =', F7.1,/,,
C      &           ' VELGAS FT/S = ',F10.3)
100  CONTINUE
200  CONTINUE

    IF (IDUCT.EQ.1 .AND. VELGAS.GT.196.85) ERR1='WARNING, GAS VELOCITY
& IN DUCT 1 EXCEEDS 60 M/S'

    IF (IDUCT.EQ.2 .AND. VELGAS.GT.196.85) ERR2='WARNING, GAS VELOCITY
& IN DUCT 2 EXCEEDS 60 M/S'

    IF (IDUCT.EQ.3 .AND. VELGAS.GT.196.85) ERR3='WARNING, GAS VELOCITY
& IN DUCT 3 EXCEEDS 60 M/S'

    IF (IDUCT.EQ.4 .AND. VELGAS.GT.196.85) ERR4='WARNING, GAS VELOCITY
& IN DUCT 4 EXCEEDS 60 M/S'

    IF (IDUCT.EQ.5 .AND. VELGAS.GT.196.85) ERR5='WARNING, GAS VELOCITY
& IN DUCT 5 EXCEEDS 60 M/S'

    IF (IDUCT.EQ.6 .AND. VELGAS.GT.196.85) ERR6='WARNING, GAS VELOCITY
& IN DUCT 6 EXCEEDS 60 M/S'

    IF (IDUCT.EQ.1 .AND. VELGAS.LT.49.21) ERR7='WARNING, GAS VELOCITY

```

& IN DUCT 1 IS LESS THAN 15 M/S'

IF (IDUCT.EQ.2 .AND. VELGAS.LT.49.21) ERR8='WARNING, GAS VELOCITY  
& IN DUCT 2 IS LESS THAN 15 M/S'

IF (IDUCT.EQ.3 .AND. VELGAS.LT.49.21) ERR9='WARNING, GAS VELOCITY  
& IN DUCT 3 IS LESS THAN 15 M/S'

IF (IDUCT.EQ.4 .AND. VELGAS.LT.49.21) ERR10='WARNING, GAS VELOCITY  
& IN DUCT 4 IS LESS THAN 15 M/S'

IF (IDUCT.EQ.5 .AND. VELGAS.LT.49.21) ERR11='WARNING, GAS VELOCITY  
& IN DUCT 5 IS LESS THAN 15 M/S'

IF (IDUCT.EQ.6 .AND. VELGAS.LT.49.21) ERR6='WARNING, GAS VELOCITY  
& IN DUCT 6 IS LESS THAN 15 M/S'

C\*\* Length the duct segment, in

LGTH(IDUCT) = EL1(IDUCT)\*DID

C\*\* MASS FOR EACH DUCT, 1bm

C\*\* TWLL, DUCT WALL THICKNESS, in

C\*\* DNWL, DUCT WALL DENSITY, 1bm/cu in

C\*\* Multiply duct mass by 1.25 accounting for flanges, bellows, etc

DMS = (PI/4.)\*(((DID+2.0\*TWL(IDUCT))\*\*2.0)-(DID\*\*2.0))  
& \*DNWL(IDUCT)\*LGTH(IDUCT)\*1.25

C\*\* MASS FOR MFI, 1bm

C\*\* NMFI, NUMBER OF MFIs

C\*\* TMFI, MFI FOIL THICKNESS, in

C\*\* DMFI, MFI FOIL DENSITY, 1bm/IN2

DMFI1 = DID+2\*TWL(IDUCT)

DMFI2 = DMFI1 + ((TMFI(IDUCT)+0.005)\*NMFI(IDUCT))

DAVG = ((DMFI1\*\*2.+DMFI2\*\*2.)/2.)\*\*0.5

MFM = PI\*DAVG\*TMFI(IDUCT)\*NMFI(IDUCT)\*LGTH(IDUCT)\*DMFI(IDUCT)

C\*\* TOTAL DUCT MASS, 1bm

TDM = DMS + MFM

C\*\* ASSIGN VARIABLES TO BE PASSED

WGTDUCT(IDUCT) = TDM

DIAM(IDUCT) = DID

VGAS(IDUCT) = VELGAS

RETURN

END

C.pa

SUBROUTINE HEXEN (MW,P,T,CP,DN,TK,VS)  
IMPLICIT REAL (A-H, K-Y)

```

C IMPLICIT INTEGER (I-J)
C COMMON / PROP / MW, P, T, CP, DN, PR, TK, VS
C
C * "THERMOPHYSICAL PROPERTIES OF NEON, ARGON, KRYPTON, AND XENON",
C   HEMISPHERE PUBLISHING CORP., 1988,(ORIGINALLY PUB IN USSR 1976)
C * MCCARTY, R.D., "THERMOPHYSICAL PROPERTIES OF HELIUM-4
C   FROM 4 - 3000 R WITH PRESSURES TO 15000 PSIA", NBS TN-622,
C   NATIONAL BUREAU OF STANDARDS, SEPTEMBER 1972.
C
C     T = TEMPERATURE (deg. F)
C     P = PRESSURE ( psia )
C
C MOLECULAR WEIGHTS, HELIUM = 4.0, XENON = 132.0
C HMW = 4.0
C XMW = 131.3
C P2=P*P
C P3=P2*P
C P4=P3*P
C P5=P4*P
C
C HELIUM-XENON HEAT CAPACITY, Btu/lbm-F
C CP = 4.97/MW
C
C HELIUM-XENON DENSITY, IDEAL GAS LAW, 1bm/cu ft
C DN = 144.0 * P * MW / (1545.35 * (T + 459.67))
C
C CALCULATE HELIUM THERMAL CONDUCTIVITY (Btu/hr-ft-deg F)
C INPUT: T = TEMPERATURE (deg. F) (-300 < T < 2540)
C        P = PRESSURE (psia) (1 < P < 300)
C
C C1 = 8.031646E-02 + 8.451721E-06*P - 1.998365E-07*P2
C & + 2.370611E-09*P3 - 1.103283E-11*P4 + 1.712007E-14*P5
C C2 = 1.207389E-04 + 2.882537E-08*P - 9.447555E-10*P2
C & + 9.894130E-12*P3 - 4.218863E-14*P4 + 6.168411E-17*P5
C C3 = - 4.119922E-08 + 2.712740E-12*P + 4.123287E-13*P2
C & - 7.577886E-15*P3 + 4.418570E-17*P4 - 7.777818E-20*P5
C C4 = 2.835525E-11 - 8.903979E-14*P + 1.830885E-15*P2
C & - 1.403159E-17*P3 + 4.103426E-20*P4 - 3.876697E-23*P5
C C5 = - 1.093462E-14 + 6.694365E-17*P - 1.532337E-18*P2
C & + 1.335025E-20*P3 - 4.706945E-23*P4 + 5.755328E-26*P5
C C6 = 1.591878E-18 - 1.349712E-20*P + 3.189740E-22*P2
C & - 2.878863E-24*P3 + 1.059138E-26*P4 - 1.354441E-29*P5
C
C HTK = C1 + C2*T + C3*T**2 + C4*T**3 + C5*T**4 + C6*T**5
C HTK = T*( T*( T*( T*( T*C6 + C5) + C4) + C3) + C2) + C1
C
C CALCULATE XENON THERMAL CONDUCTIVITY (Btu/hr-ft-F)
C INPUT: T = TEMPERATURE (deg. F) (-150 < T < 1900)
C        P = PRESSURE (psia) (14.5 < P < 300)
C
C C1 = 2.755341E-03 + 1.455947E-06*P + 1.415970E-09*P2
C C2 = 5.708226E-06 - 3.011999E-09*P - 9.272117E-12*P2
C C3 = - 8.638393E-10 + 5.376902E-12*P + 1.969676E-14*P2

```

```

C4 = - 1.315491E-13 - 5.839467E-15*p - 1.884432E-17*p^2
C5 = 1.574533E-16 + 3.108693E-18*p + 8.322497E-21*p^2
C6 = - 3.158769E-20 - 6.218946E-22*p - 1.375837E-24*p^2
C
C XTK = C1 + C2*T + C3*T**2 + C4*T**3 + C5*T**4 + C6*T**5
C XTK = T*( T*( T*( T*C6 + C5) + C4) + C3) + C2) + C1
C
C TA1 = (1.0 + SQRT(HTK/XTK)*(HMW/XMW)**0.25)**2.0 /
+ (2.82843*SQRT(1.0 + HMW/XMW))
C TA2 = (1.0 + SQRT(XTK/HTK)*(XMW/HMW)**0.25)**2.0 /
+ (2.82843*SQRT(1.0 + HMW/XMW))
C TB1 = (2.41*(HMW - XMW)*(HMW - 0.142*XMW))/(HMW + XMW)**2.0
C TB2 = (2.41*(XMW - HMW)*(XMW - 0.142*HMW))/(HMW + XMW)**2.0
C TC1 = TA1*(1.0 + TB1)
C TC2 = TA2*(1.0 + TB2)
C MIX THERMAL CONDUCTIVITY
C TK = HTK/(1.0+TC1*(MW-HMW)/(XMW-MW)) +
+ XTK/(1.0+TC2*(XMW-MW)/(MW-HMW))
C
C CALCULATE HELIUM DYNAMIC VISCOSITY (lbm/hr-ft)
C INPUT: T = TEMPERATURE (deg. F) (-300 < T < 2540)
C P = PRESSURE (psia) (1 < P < 300)
C
C1 = 4.314593E-02 + 6.194756E-08*p + 4.998798E-09*p^2
& - 1.323100E-10*p^3 + 1.099206E-12*p^4 - 2.282117E-15*p^5
C2 = 6.446683E-05 - 4.772369E-09*p + 5.663575E-11*p^2
& - 6.687830E-13*p^3 + 2.848734E-15*p^4 - 4.006185E-18*p^5
C3 = - 1.809663E-08 + 1.830709E-11*p - 3.212607E-13*p^2
& + 4.683985E-15*p^3 - 2.644519E-17*p^4 + 4.624817E-20*p^5
C4 = 9.059456E-12 - 2.194842E-14*p + 4.340343E-16*p^2
& - 6.655726E-18*p^3 + 3.952066E-20*p^4 - 7.122290E-23*p^5
C5 = - 2.669536E-15 + 1.039065E-17*p - 2.189294E-19*p^2
& + 3.441683E-21*p^3 - 2.091342E-23*p^4 + 3.819015E-26*p^5
C6 = 3.128055E-19 - 1.690270E-21*p + 3.704323E-23*p^2
& - 5.910116E-25*p^3 + 3.639098E-27*p^4 - 6.694850E-30*p^5
C
C HVIS = C1 + C2*T + C3*T**2 + C4*T**3 + C5*T**4 + C6*T**5
C HVIS = T*( T*( T*( T*C6 + C5) + C4) + C3) + C2) + C1
C
C CALCULATE XENON DYNAMIC VISCOSITY (lbm/hr-ft)
C INPUT: T = TEMPERATURE (deg. F) (-10 < T < 2000)
C P = PRESSURE (psia) (14.51 < P < 290.1)
C
C1 = 4.877915E-02 + 8.373816E-06*p + 1.661438E-08*p^2
C2 = 1.000932E-04 + 1.584259E-09*p - 1.335857E-10*p^2
C3 = - 1.397578E-08 - 2.113691E-11*p + 3.380712E-13*p^2
C4 = - 4.222308E-12 + 2.842605E-14*p - 3.811784E-16*p^2
C5 = 4.001660E-15 - 1.547142E-17*p + 1.955617E-19*p^2
C6 = - 8.303670E-19 + 3.027289E-20*p - 3.717919E-23*p^2
C
C XVIS = C1 + C2*T + C3*T**2 + C4*T**3 + C5*T**4 + C6*T**5
C XVIS = T*( T*( T*( T*C6 + C5) + C4) + C3) + C2) + C1

```

```

C
  VIS1 = (1.0 + SQRT(HVIS/XVIS) * (XMW/HMW)**0.25)**2.0 /
+          ( 2.82843*SQRT(1.0 + HMW/XMW))
  VIS2 = VIS1*XVIS/HVIS*(HMW/XMW)
C  MIX VISCOSITY, (lbm/hr-ft)
  VS = HVIS/(1.0+VIS1*(MW-HMW)/(XMW-MW)) +
+      XVIS/(1.0+VIS2*(XMW-MW)/(MW-HMW))
C
C  MIX PRANDTL NUMBER, He-Xe
  PR = VS*CP/TK
  RETURN
  END
C.pa
SUBROUTINE XNAK(T, CP, DN, TK ,VS)
C** THERMAL PROPERTIES OF NaK LIQUID
C**   T - INPUT TEMPERATURE (deg-R)
C**   CP - SPECIFIC HEAT (Btu/lbm-R)
C**   DN - DENSITY (lbm/cu-ft)
C**   TK - THERMAL CONDUCTIVITY (Btu/hr-ft-R)
C**   VS - DYNAMIC VISCOSITY (lbm/hr ft)
C** IMPLICIT REAL*8 (A-H,K-Y)

CP = 0.26478 -0.000089*T +4.093060E-08*T**2.0 -4.532164E-12*T**3.0
DN = 58.54299-(0.008208*T)
C** VS - DYNAMIC VISCOSITY (lbm/ft-s)
VS = 0.000822-1.142435E-6*T+6.125737E-10*T**2.0-1.13018E-13*T**3.0
C** VS - DYNAMIC VISCOSITY (lbm/hr ft)
VS = VS*3600.0
TK = 7.313351+0.013983*T -7.660423E-06*T**2.0 +1.189370E-09*T**3.0
RETURN
END

SUBROUTINE TAC
C **** TAC is checked-out and final on October 6, 1992

IMPLICIT REAL (A-H)
IMPLICIT INTEGER (I-J)
IMPLICIT REAL (K-Z)
REAL JCON

CHARACTER*10 GAS(2),CYCDES(8),TITLER(8),TVAR(18),
& COMPTYPE,TURBTYP,CLNTTYPE
CHARACTER*20 GENTYPE,INTTYPE
CHARACTER*64 ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNINGT,WARNINGC,
& WARNINGR,WARNINGG

COMMON/DIAGNOS/ERRORT,ERRORC,ERRORM,ERRORG,ERRORF,WARNINGT,
& WARNINGC,WARNINGR,WARNINGG

COMMON/OUTP/MASSTOT,MASSTAC,RECLC,RECLOA,RECH,RECW
COMMON/DUCT/ MDCTTOT, MASSDCT(6),DIADUCT(6)

```

```

COMMON/HSACALCS/ QHSA,MASSHSA,DP9
COMMON/CONFIG/COMPTYPE,TURBTYPE,GENTYPE,INTTYPE,CLNTTYPE

COMMON/ALTERNTR/DGENRTR,DGENSTR,LGENTOT,MASSGEN,TIPSPDG,COE,
& ETAGEN,COOLING,WCLNT,VOLTAGE,KVA,GENASP,TINCLNT,TOUTCLNT,
& CPCLNT,LIFETIME
COMMON/AERODYN/ETACOMP,COMPDIA,UTIPC,RADHC,RAD1C,
& RADNC,ANSQCL,COMPSS,ISTGC,ETATURB,TURBDIA,UTIPT,
& RADHT,RADIT,RADNT,ANSQT,ANSQTL,TURBSS,ISTGT

COMMON/OPTIM/ VAR(18),TVAR,IPRINT
COMMON/FLUID/ GAS,GAMMA,MOLWT,CP,PRNDTL
COMMON/TITLE/ CYCDES,TITLER
COMMON/LOSS/ QLOSS8,QLOSS9,QLOSS10,QLOSS13,FCTQ8,FCTQ10,FCTQ13
COMMON/CYCLE/TEMP(17),PRESS(17),FLOW(17),PR(17),BETA,COMFLO,
& PRC,EFFC,EFFT,EFFR,EFFA,XBLC,XBLT,SPEED,POWER,PWRFCCTR,
& DPOP6,DPOP7,DPOP8,DPOP9,DPOP10,DPOP13,DPOP14,DPOP15,
& DPOP16,DPOP17,NETSP,GROSSEP,EFFCYCLE,WINDAGE
COMMON/GASBRG/BRGLOSS
COMMON/RECUP/ DPOPREC,UAML,UAMH,DPML,DPMH,
& PREXP,NFINL,TFINL,SPACEL,LEQL,ROFINL,KFINL,NFINH,TFINH,SPACEH,
& LEQH,ROFINH,KFINH,TPLATE,ROPLAT,TBRAZE,ROBRAZ,
& SIGMAL,ALPHAL,AFINL,APlATL,HDL,SIGMAH,ALPHAH,AFINH,APlATH,HDH,
& PITCH,MASSREC
COMMON/MASSES/MASSCMP,MASSTRB,MWHLIC,MWHLT,MSFTC,MSFTT,MCASEC,
& MCASET,MDIAC,MDIAT

COMMON/CONST/PI,RU,GO,JCON,RZERO

```

RHOTZM=.37  
RHOTI=.164

C \*\*\*\* This routine was formulated by Allied-Signal. It predicts  
C TAC mass based on dimensional and thermodynamic values for both  
C axial and radial based machines. Mass includes compressor  
C turbine, and generator masses with allowances for packaging and  
C \*\*\*\* structural mass.  
C

C IF (COMPTYPE.EQ.'AXIAL ') THEN  
C \*\*\*\* Mass estimate for an axial compressor with dimensions predicted  
C \*\*\*\* by Subroutine AXCOMP; All Ti alloy construction assumed.

```

UHUBC=UTIPC*RADHC/RAD1C
AR=2.
MWHLIC=0.
CXSUM=0.
HSUM=0.

```

C \*\*\*\* Compute and sum the wheel masses based on GFSD estimate algorithms
DO 100, I=1,ISTGC
DTIPC=2./12.\*(RAD1C+(RADNC-RAD1C)\*(FLOAT(I)-.5)/FLOAT(ISTGC))
H=DTIPC\*12./2.-RADHC

```

CX=MAX(.5,H/AR)
CXSUM=CXSUM+CX
HSUM=HSUM+H
IF (UHUBC**2.LT.750000.) THEN
  MWHLC=DTIIPC**3*(1.7E-5*UHUBC**2-2.1)+MWHLC
ELSE
  MWHLC=DTIIPC**3*(8.5E-5*UHUBC**2-53.4)+MWHLC
END IF
C   WRITE (*,445) I,ISTGC,UTIIPC,RADHC,RAD1C,RADNC,MWHLC,DTIIPC,UHUBC
C   WRITE (61,445) I,ISTGC,UTIIPC,RADHC,RAD1C,RADNC,MWHLC,DTIIPC,UHUBC
C 445 FORMAT (' I,ISTGC,UTIIPC,RADHC,RAD1C,RADNC,MWHLC,DTIIPC,UHUBC '
C   &/1X,2I3,3E12.5/1X,5E12.5)
100 CONTINUE
  CXAVG=CXSUM/FLOAT(ISTGC)
  HAVG=HSUM/FLOAT(ISTGC)
C **** Estimate the shaft mass
C **** Compute blade heights
  TAU=50000.
  TORQUE=FLOW(1)*CP*(TEMP(2)-TEMP(1))*JCON/(SPEED*PI/30.)
  RSHAFT=(2.*TORQUE*12./(PI*TAU))**.3333
  MSFTC=RHOI*(PI*RSHAFT**2)*CXAVG*(9.+3.*FLOAT(ISTGC-1)+12.)
C   WRITE (*,446) CXAVG,HAVG,TORQUE,RSHAFT,MSFTC
C   WRITE (61,446) CXAVG,HAVG,TORQUE,RSHAFT,MSFTC
C 446 FORMAT (' CXAVG,HAVG,TORQUE,RSHAFT,MSFTC'/1X,5E12.5)
C **** Compute case mass
  LCASE=CXAVG*(15.+3.*FLOAT(ISTGC-1))
  RCASE=(RADNC+RAD1C)/2.
  TCASE=MAX(.1,(PRESS(1)+PRESS(4))*RCASE/(2.*50000.))
  VCASE=2.*PI*(RCASE+TCASE/2.)*LCASE*TCASE
  MCASEC=1.15*VCASE*RHOI
C   WRITE (*,447) LCASE,RCASE,TCASE,VCASE,MCASEC
C   WRITE (61,447) LCASE,RCASE,TCASE,VCASE,MCASEC
C 447 FORMAT (' LCASE,RCASE,TCASE,VCASE,MCASEC'/1X,5E12.5)

C **** Compute diaphragm mass
  VOL1=2.*PI*RADNC*.1*(CX+.8+2.*1.)
  VOL2=2.*PI*RADHC*(CX+H)*.2
  VOL3=2.*PI*RADHC*(CX+1.)*CX/5.
  A4=(RADHC-RSHAFT)*CX/8.
  R4=RSHAFT+(RADHC-RSHAFT)/2.
  VOL4=2.*PI*A4*R4+21./8.*CX**2*(RADHC-RSHAFT)
  VOL5=2.*PI*RSHAFT*(CX+1.)*CX/8.
  MDIAC=RHOI*(VOL1+VOL2+VOL3+VOL4+VOL5)
  MASSCMP=MWHLC+MSFTC+MCASEC+MDIAC
C   WRITE (*,448) VOL1,VOL2,VOL3,VOL4,VOL5,MDIAC,MASSTRB
C   WRITE (61,448) VOL1,VOL2,VOL3,VOL4,VOL5,MDIAC,MASSTRB
C 448 FORMAT (' VOL1,VOL2,VOL3,VOL4,VOL5,MDIAC,MASSTRB'2(/1X,5E12.5))

  ELSE
C **** Mass estimate for an radial compressor with dimensions predicted
C **** by Subroutine RADCOMP; Based on average of GFSD designs.

```

```

MASSCMP=30.*(1.+5.)*(COMPDIA/12.)**3

ENDIF

IF (TURBTYP.EQ.'AXIAL      ') THEN
C **** Mass estimate for an axial turbine with dimensions predicted
C **** by Subroutine AXTURB; All refractory alloy construction assumed.

UHUBT=UTIPT*RADHT/RADNT
AR=2.
MWHLT=0.
CXSUM=0.
HSUM=0.
C **** Compute and sum the WHLT masses based on GFSD estimate algorithms
DO 110, I=1,ISTGT
DTIPT=2./12.*((RAD1T+(RADNT-RADIT)*(FLOAT(I)-.5)/FLOAT(ISTGT)))
H=DTIPT*12./2.-RADHT
CX=MAX(.5,H/AR)
CXSUM=CXSUM+CX
HSUM=HSUM+H
IF (UHUBT**2.LT.750000.) THEN
MWHLT=DTIPT**3*(1.7E-5*UHUBT**2-2.1)+MWHLT
ELSE
MWHLT=DTIPT**3*(8.5E-5*UHUBT**2-53.4)+MWHLT
END IF
C      WRITE (*,345) I,ISTGT,UTIPT,RADHT,RAD1T,RADNT,MWHLT,DTIPT,UHUBT
C      WRITE (61,345) I,ISTGT,UTIPT,RADHT,RAD1T,RADNT,MWHLT,DTIPT,UHUBT
C 345 FORMAT (' I,ISTGT,UTIPT,RADHT,RAD1T,RADNT,MWHLT,DTIPT,UHUBT '
C &/1X,2I3,3E12.5/1X,5E12.5)
110 CONTINUE
CXAVG=CXSUM/FLOAT(ISTGT)
HAVG=HSUM/FLOAT(ISTGT)
C **** Estimate the shaft mass
C **** Compute blade heights
TAU=50000.
TORQUE=FLOW(10)*CP*(TEMP(10)-TEMP(12))*JCON/(SPEED*PI/30.)
RSHAFT=(2.*TORQUE*12./(PI*TAU))**.3333
MSFTT=RHOTZM*(PI*RSHAFT**2)*CXAVG*(9.+3.*FLOAT(ISTGT-1)+12.)
C      WRITE (*,346) CXAVG,HAVG,TORQUE,RSHAFT,MSFTT
C      WRITE (61,346) CXAVG,HAVG,TORQUE,RSHAFT,MSFTT
C 346 FORMAT (' CXAVG,HAVG,TORQUE,RSHAFT,MSFTT'/1X,5E12.5)
C **** Compute case mass
LCASE=CXAVG*(15.+3.*FLOAT(ISTGT-1))
RCASE=(RADNT+RAD1T)/2.
TCASE=MAX(.1,(PRESS(10)+PRESS(12))*RCASE/(2.*20000.))
VCASE=2.*PI*(RCASE+TCASE/2.)*LCASE*TCASE
MCASET=1.15*VCASE*.286
C      WRITE (*,347) LCASE,RCASE,TCASE,VCASE,MCASET
C      WRITE (61,347) LCASE,RCASE,TCASE,VCASE,MCASET
C 347 FORMAT (' LCASE,RCASE,TCASE,VCASE,MCASET'/1X,5E12.5)

```

```

C **** Compute diaphragm mass
VOL1=2.*PI*RADNT*.1*(CX+.8+2.*1.)
VOL2=2.*PI*RADHT*(CX+H)*.2
VOL3=2.*PI*RADHT*(CX+1.)*CX/5.
A4=(RADHT-RSHAFT)*CX/8.
R4=RSHAFT+(RADHT-RSHAFT)/2.
VOL4=2.*PI*A4*R4+21./8.*CX**2*(RADHT-RSHAFT)
VOL5=2.*PI*RSHAFT*(CX+1.)*CX/8.
MDIAT=RHOTZM*(VOL1+VOL2+VOL3+VOL4+VOL5)
MASSTRB=MWHLT+MSFTT+MCASET+MDIAT
C      WRITE (*,348) VOL1,VOL2,VOL3,VOL4,VOL5,MDIAT,MASSTRB
C      WRITE (61,348) VOL1,VOL2,VOL3,VOL4,VOL5,MDIAT,MASSTRB
C 348 FORMAT (' VOL1,VOL2,VOL3,VOL4,VOL5,MDIAT,MASSTRB'2(/1X,5E12.5))

      ELSE

C **** Mass estimate for an radial turbine with dimensions predicted
C **** by Subroutine RADTURB; Based on average of GFSD designs.

      MASSTRB=46.*(RHOTZM/.286+5.)*(TURBDIA/12.)**3

      ENDIF

C **** Base weight of the TAC = MASSTRB + MASSCMP + MASSGEN
      MASSTAC=MASSCMP+MASSTRB+MASSGEN

C **** Allowing an additional 20% for bearing system and interface:
      MASSTAC=MASSTAC*1.2

      RETURN
      END

```

**Appendix C**  
**Input Data File Listing**

CYC\$INP - Input data file for cycle data

500 kWe CBC Test Case for NASA T018 Code - 5 December 92

1	500.	.9	1400.	2.5	1144.44	
RADIAL	RADIAL	RING WOUND	TPTL	PMG	TRANSFORMER	DOWTHERM A
XENON	HELIUM	.5	511.11	522.22	.0	.0
5.	.005	.01	.01			
T1, K	PRC	MW	ER	NSC	N, RPM	DP/P6 DP/PREC
DP/P8	DP/P9	DP/P10	DP/P13	DP/P15	DP/P16	DP/P17
375.	1.8	20.	.85	42.	28000.	.005 .02
.005	.002	.003	.003	.005	.005	.005

REC\$INP - Recuperator input data file

Plate Fin Recuperator; Kays & London Matrix, Exponent=.615

.615	.15	.15	.15	.15	
6.3	.0127	.3175	.1524	8.03	22.14
7.87	.0127	.254	.1206	8.03	22.14
	.0254			8.03	
	.00381			8.86	

IHX\$INP - Intermediate heat exchanger input data file

1	226.8	1166.7	1111.1	.102	6.0
14.4	.025	8.34	8.34	1.27	

DUCT\$INP - Ducting system input data file

15.0	15.0	15.0	15.0	15.0	15.0
60.0	60.0	60.0	60.0	60.0	60.0
0.102	0.102	0.102	0.102	0.102	0.102
4.43	8.30	8.30	8.30	4.43	4.43
0.00	100.0	100.0	100.0	0.0	0.0
0.0013	0.0013	0.0013	0.0013	0.0013	0.0013
5.54	5.54	5.54	5.54	5.54	5.54
0.001	0.001	0.001	0.001	0.001	0.001

**Appendix D**  
**Example Output [PROG\$OUT]**  
**500 kWe, Axial Case**

## \*\*\* PARAMETRIC DATA \*\*\*\*

T1, K	.37500E+03	PRC	.30000E+01	MW	.20000E+02
ER	.00000E+00	NSC	.42000E+02	N, RPM	.25000E+05
DP/P6	.50000E-02	DP/PREC	.20000E-01	DP/P8	.00000E+00
DP/P9	.11000E-01	DP/P10	.60000E-02	DP/P13	.70000E-02
DP/P15	.00000E+00	DP/P16	.11000E-01	DP/P17	.60000E-02

## \*\*\* MASS and EFFICIENCY DATA \*\*\*\*

TAC MASS, KG	221.	REC MASS, KG	0.
DCTNG MASS, KG	117.		
CMP MASS, KG	53.	TRB MASS, KG	54.
GEN MASS, KG	78.	TOT MASS, KG	221.
CYC EFF	.20364	ETA COMPR	.81346
ETA TURB	.93434	ETA ALT	.96620
BETA	.95486	GROSS EP kWe	500.0

## \*\*\* DIMENSIONAL DATA \*\*\*\*

## \*\*\*\*\* AXIAL COMPRESSOR GEOMETRY

NO STAGES	7	HUB RAD, cm	11.065	TIP R INL,cm	13.470
TIP R DCH,cm	12.419	AN^2C	.180E+11	AN^2LIMIT	.405E+11
SPEC SPD	42.0	MASS, kg	52.7		

## \*\*\*\*\* AXIAL TURBINE GEOMETRY

NO STAGES	2	HUB RAD, cm	12.411	TIP R INL,cm	14.741
TIP R DCH,cm	15.851	AN^2T	.296E+11	AN^2LIMIT	.385E+11
SPEC SPD	44.4	MASS, kg	53.8		

## \*\*\*\*\* GENERATOR GEOMETRY AND PARAMETERS

Rotor OD, cm	11.344	Stator OD,cm	22.895	Length, cm	32.792
Rotor L/D	2.50	Mass, kg	77.55	Tip Spd m/s	148.
kVA	555.6	Pwr Factr	.900	Volts,l-1	1400.
CInt Tin, K	511.	CInt Tout, K	522.	CInt Flow,kg/s	1.21

TYPE= RING WOUND TPTL PMG INTERFACE= TRANSFORMER

COOLANT= DOWTHERM A

## \*\*\* THERMODYNAMIC DATA \*\*\*\*

TEMPERATURES, K	375.00	629.40	629.40	629.40	629.40	629.40
TEMPERATURES, K	629.40	626.81	1144.44	1139.26	1139.26	773.53
TEMPERATURES, K	768.35	768.35	768.35	375.00	375.00	
PRESSES, KPA	253.28	506.56	506.56	759.84	759.84	756.04
PRESSES, KPA	756.04	756.04	747.73	743.24	501.35	259.46
PRESSES, KPA	257.64	257.64	257.64	254.81	253.28	
PRESS RATIOS	1.000000	1.000000	1.000000	1.000000	1.000000	.995000
PRESS RATIOS	1.000000	1.000000	.989000	.994000	1.000000	1.000000
PRESS RATIOS	.993000	1.000000	1.000000	.989000	.994000	
FLOW RATES, KG/S	4.57	4.57	4.57	4.57	4.57	4.57
FLOW RATES, KG/S	4.57	4.57	4.57	4.57	4.57	4.57
FLOW RATES, KG/S	4.57	4.57	4.57	4.57	4.57	4.57

## \*\*\* PARASITIC LOSSES \*\*\*\*

THERMAL LOSSES [kW]- HSA		.00	THTML 8	-12.28
	THRML 10	-24.55	THRML 13	-24.55
SHAFT LOSSES [kW]- TOT BRGS		10.00	WINDAGE	.64

## \*\*\*\*\* Ducting Configuration \*\*\*\*\*

Duct 1 diameter, cm = 20.19038 Length, cm= 302.85570  
 Duct 2 diameter, cm = .00000 Length, cm= .00000  
 Duct 3 diameter, cm = 22.54260 Length, cm= 225.42600  
 Duct 4 diameter, cm = 33.20548 Length, cm= 332.05480  
 Duct 5 diameter, cm = .00000 Length, cm= .00000  
 Duct 6 diameter, cm = 28.48324 Length, cm= 284.83240

Duct 1 velocity, m/s= 49.09991 Weight, kg= 10.90493  
 Duct 2 velocity, m/s= .00000 Weight, kg= .00000  
 Duct 3 velocity, m/s= 72.78912 Weight, kg= 28.93446  
 Duct 4 velocity, m/s= 65.33964 Weight, kg= 62.40010  
 Duct 5 velocity, m/s= .00000 Weight, kg= .00000  
 Duct 6 velocity, m/s= 43.82466 Weight, kg= 14.44725  
 WARNING, GAS VELOCITY IN DUCT 3 EXCEEDS 60 M/S  
 WARNING, GAS VELOCITY IN DUCT 4 EXCEEDS 60 M/S

## IHX Design Parameters

XDPSHELL1	- shell side pressure drop, kPa	7.949
ANTUBES1	- number of heat exchanger tubes	195.
XDPTUBE1	- tube side pressure drop, kPa	3.289
XDOTL21	- diameter of the shell, cm	37.3
XALSHEL1	- length of the shell, cm	261.0
XAMSHELL1	- mass of the shell, kg	70.2
XAMPLATES1	- mass of the plates, kg	3.9
XAMTUBES1	- mass of the tubes, kg	158.6
XAMINSUL1	- mass of the insulation, kg	8.2
XAMHEADS1	- mass of the heads, kg	8.8
XAMTSHT1	- mass of the tubesheets, kg	8.8
XAMSTR1	- structure and brackets, kg	28.5
XANETMASS1	- net shell and tube unit, kg	271.4
XXMNHEX1	- mass of tubeside fluid, kg	43.1
XHSHELL1	- shell side h, W/m <sup>2</sup> -K	1631.4
AFRIC1	- friction factor	.091
XUNEW1	- overall U, W/m <sup>2</sup> -K	1422.72400
RETUBE1	- tubeside Reynolds number	26973.310
XTHC1	- tubeside h, W/m <sup>2</sup> -K	52423.4
PRATIO	- tube pitch ratio	1.8

**Appendix E**  
**Example Output [PROG\$OUT]**  
**500 kWe, Radial Case**

500 kWe CBC Test Case 5 December 92 (July 30, 1993 test case)  
 Plate Fin Recuperator; Kays & London Matrix, Exponent=.615

(1/4)

```
*** PARAMETRIC DATA ****
T1, K          .37500E+03   PRC      -18000E+01   MW      .20000E+02
ER             .85000E+00   NSC      -.45000E+02   N, RPM   .28000E+05
DP/P6          .50000E-02   DP/PREC   .20000E-01   DP/P8     .50000E-02
DP/P9          .20000E-02   DP/P10    -.30000E-02   DP/P13   .30000E-02
DP/P15         .50000E-02   DP/P16    -.50000E-02   DP/P17   .50000E-02

*** MASS and EFFICIENCY DATA ****
TAC MASS, KG   382.      REC MASS, KG   252.
DCTNG MASS, KG 112.      TRB MASS, KG   166.
CNP MASS, KG   74.       PCS MASS, KG   747.
GEN MASS, KG   79.       ETA COMPR    .79093
CYC EFF        .26309
ETA TURB       .92081   ETA ALT     .96620
BETA           .94612   GROSS IP kWe  500.0

*** DIMENSIONAL DATA ****
**** RADIAL COMPRESSOR GEOMETRY
COMP DIA, cm   29.660   TIP SPD., m/s  432.  SPEC SP   45.000
COMP MASS, kg   73.7

**** RADIAL TURBINE GEOMETRY
TURB DIA, cm   32.959   TIP SPD., m/s  483.  SPEC SP   53.265
TURB MASS, kg   166.2

**** GENERATOR GEOMETRY AND PARAMETERS
Rotor OD, cm   11.768   Stator OD, cm  22.367   Length, cm  33.978
Rotor L/D       2.50     Mass, kg       78.79    Tip Spd m/s  173.
kVA            555.6    Pwr Factor   .900    Volts, l-l   1400.

Frequency, Hz   466.667
Clnt Tin, K     511.    Clnt Tout, K   522.    Clnt Flow,kgs/s  1.32
TYPE= RING WOUND TPTL PMG INTERFACE= TRANSFORMER
COOLANT= Dowtherm A

*** THERMODYNAMIC DATA ****
TEMPERATURES, K 375.00   500.67   500.67   500.67   500.67
TEMPERATURES, K 871.19   869.82   1144.44   1141.69   939.33
TEMPERATURES, K 936.58   566.05   566.05   375.00   375.00
PRESSURES, KPA 1339.18   1874.86   1874.86   2410.53   2398.48
PRESSURES, KPA 2379.17   2367.28   2362.54   2355.46   1867.82
PRESSURES, KPA 1376.05   1359.47   1352.68   1345.91   1339.18
PRES RATIOS    1.000000  1.000000  1.000000  1.000000  1.000000
PRES RATIOS    .991951  .995000  .998000  .997000  1.000000
PRES RATIOS    .997000  .987952  .995000  .995000  .995000
FLOW RATES, KG/S 6.66     6.66     6.66     6.66     6.66
FLOW RATES, KG/S 6.66     6.66     6.66     6.66     6.66
FLOW RATES, KG/S 6.66     6.66     6.66     6.66     6.66
*** PARASITIC LOSSES ****
THERMAL LOSSES [kW]- HSA   .00     THML 8   -9.50
SHAFT LOSSES [kW]- THML 10  -19.01   THRL 13   -19.01
                           10.00   WINDAGE  3.30
```

## 500 kWe CBC Test Case 5 December 92 (July 30, 1993 test case)

## INPUT GEOMETRY FOR COUNTERFLOW HEAT EXCHANGER DESIGN PROGRAM

(2/4)

## Plate Fin Recuperator; Kays &amp; London Matrix, Exponent=.615

	GAS	GAS UA MARGIN (- - -)	DP MARGIN (- - -)	COND. FACTOR (- - -)
L.P. SIDE ----- H.P. SIDE -----	XENON XENON	HELIUM HELIUM	.1500 .1500	.1500 .1500
	NUMBER (PER CM)	THICKNESS (CM)	SPACING (CM)	EQ. LENGTH (CM)
L.P. SIDE FINS ----- H.P. SIDE FINS ----- SPLITTER PLATES ----- BRAZE MATERIAL -----	6.30 7.87	.013 .013 .025 .004	.317 .254	.152 .121
				8.030 8.030 8.030 8.860

	FLOW AREA RATIO (- - -)	SURFACE AREA/VOLUME AREA/(CM) (- - -)	FIN AREA/VOLUME (1/CM) (- - -)	PLATE AREA/VOLUME (1/CM) (- - -)	HYDRAULIC DIAMETER (CM) (- - -)
L.P. SIDE FINS ----- H.P. SIDE FINS -----	.45061 .34900	9.128 8.996	6.171 6.103	2.957 2.893	.19746 .15518

LOW PRESSURE SIDE				HIGH PRESSURE SIDE			
	GAS IS XENON	HELIUM		GAS IS XENON	HELIUM		
EFFECTIVENESS (----)	INLET TEMPERATURE (DEG. K)	EXIT TEMPERATURE (DEG. K)	*	EFFECTIVENESS (----)	INLET TEMPERATURE (DEG. K)	EXIT TEMPERATURE (DEG. K)	*
.85000E+00	.93658E+03	.56605E+03	*	.85000E+00	.50067E+03	.87119E+03	*
FLOW RATE (KG/SEC)	INLET PRESSURE (KPA),	PRESSURE DROP (DP/P)	*	FLOW RATE (KG/SEC)	INLET PRESSURE (KPA),	PRESSURE DROP (DP/P)	*
.66624E+01	.13761E+04	.86618E-02	*	.66624E+01	.23985E+04	.53782E-02	*
FIN EFFECTIVENESS (----)	REYNOLDS NO. (----)	PRANDTL NO. (----)	*	FIN EFFECTIVENESS (----)	REYNOLDS NO. (----)	PRANDTL NO. (----)	*
.61557E+00	.21580E+04	.26459E+00	*	.66553E+00	.23368E+04	.26376E+00	*
H.T. CONDUCTANCE (W/ DEG.K)	VISCOSITY (CP)	SPECIFIC HEAT (CAL/GM-DEG.K)	*	H.T. CONDUCTANCE (W/ DEG.K)	VISCOSITY (CP)	SPECIFIC HEAT (CAL/GM-DEG.K)	*
.80014E+05	.50269E-01	.24818E+00	*	.10328E+06	.47114E-01	.24818E+00	*
HEADER WEIGHT (KG)	FIN WEIGHT (KG)	FLUID WEIGHT (KG)	*	HEADER WEIGHT (KG)	FIN WEIGHT (KG)	FLUID WEIGHT (KG)	*
.00000E+00	.44082E+02	.00000E+00	*	.00000E+00	.43762E+02	.00000E+00	*
TOTAL WEIGHT (KG)	PLATE WEIGHT (KG)	SPACER BAR WEIGHT (KG)	*	WRAP UP WEIGHT (KG)	BRAZE WEIGHT (KG)	(----)	*
.25232E+03	.60382E+02	.00000E+00	*	.00000E+00	.19987E+02	.00000E+00	*
CORE LENGTH (CM)	CORE WIDTH (CM)	CORE HEIGHT (CM)	*	OVERALL LENGTH (CM)	VOLUME (CU.CM)	TOTAL PRESSURE DROP (DP/P)	*
.34231E+02	.51875E+02	.51875E+02	*	.34231E+02	.00000E+00	.20000E-01	*

## \*\*\*\*\* Ducting Configuration \*\*\*\*\*

Duct 1 diameter, cm = 12.91509 Length, cm= 193.72630  
 Duct 2 diameter, cm = 14.96846 Length, cm= 224.52680  
 Duct 3 diameter, cm = 18.38337 Length, cm= 275.75050  
 Duct 4 diameter, cm = 22.70933 Length, cm= 340.64000  
 Duct 5 diameter, cm = 17.61942 Length, cm= 264.29120  
 Duct 6 diameter, cm = 15.94143 Length, cm= 239.42220

Duct 1 velocity, m/s= 43.89671 Weight, kg= 4.47463  
 Duct 2 velocity, m/s= 57.62262 Weight, kg= 19.32025  
 Duct 3 velocity, m/s= 50.54087 Weight, kg= 28.98738  
 Duct 4 velocity, m/s= 46.53008 Weight, kg= 44.03994  
 Duct 5 velocity, m/s= 47.28386 Weight, kg= 8.31066  
 Duct 6 velocity, m/s= 38.64503 Weight, kg= 6.80727

## IMX Design Parameters

XDPSHELL1	- shell side pressure drop, kPa	4.536
ANTUBES1	- number of heat exchanger tubes	191.
XDTUBE1	- tube side pressure drop, kPa	2.052
XDOTL21	- diameter of the shell, cm	35.1
XALSHLL1	- length of the shell, cm	245.5
XAMSHLL1	- mass of the shell, kg	155.1
XAMPLATES1	- mass of the plates, kg	8.6
XAMTUBES1	- mass of the tubes, kg	146.2
XAMINSUL1	- mass of the insulation, kg	7.2
XAMHEADS1	- mass of the heads, kg	19.4
XAMTSHT1	- mass of the tubesheets, kg	19.4
XAMSTR1	- structure and brackets, kg	17.8
XANETMASS1	- net shell and tube unit, kg	373.6
XXMMHEX1	- mass of tubeside fluid, kg	37.8
XHSHELL1	- shell side h, W/m <sup>2</sup> -K	2365.6
AFRIC1	- friction factor	.093
XUNEM1	- overall U, W/m <sup>2</sup> -K	1945.99900
RETUBE1	- tubeside Reynolds number	23297.970
XTHC1	- tubeside h, W/m <sup>2</sup> -K	49241.9
PRATIO	- tube pitch ratio	1.7

## SUMMARY MASS TABLE

TAC Mass, kg	382.443
Recuperator Mass, kg	252.321
Ducting mass, kg	111.940

Power Conversion System, kg	746.704
Intermediate heat exchanger (wet), kg	411.379



<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.			
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED	
	November 1993	Final Contractor Report	
4. TITLE AND SUBTITLE		5. FUNDING NUMBERS	
Brayton Power Conversion System Parametric Design Modelling for Nuclear Electric Propulsion		WU-000-00-00-00 NAS3-25808 Task Order 18	
6. AUTHOR(S)		7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	
Thomas L. Ashe and William D. Otting		Rockwell International Corporation Rocketdyne Division 6633 Canoga Avenue Canoga Park, California 91303	
8. PERFORMING ORGANIZATION REPORT NUMBER		9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	
E-8345		National Aeronautics and Space Administration Washington, DC 20546-0001	
10. SPONSORING/MONITORING AGENCY REPORT NUMBER		11. SUPPLEMENTARY NOTES	
NASA CR-191135		Thomas L. Ashe, Allied-Signal Aerospace Company, Tempe, Arizona 85282; and William D. Otting, Rocketdyne, Canoga Park, California 91303. Project Manager, Robert L. Cataldo, Advanced Space Analysis Office, NASA Lewis Research Center, organization code 6840, 216-977-7082.	
12a. DISTRIBUTION/AVAILABILITY STATEMENT		12b. DISTRIBUTION CODE	
Unclassified - Unlimited Subject Category: 44  Available electronically at <a href="http://gltrs.grc.nasa.gov/GLTRS">http://gltrs.grc.nasa.gov/GLTRS</a> This publication is available from the NASA Center for AeroSpace Information, 301-621-0390.			
13. ABSTRACT (Maximum 200 words)			
The parametrically based closed Brayton cycle (CBC) computer design model was developed for inclusion into the NASA LeRC overall Nuclear Electric Propulsion (NEP) end-to-end systems model. The code is intended to provide greater depth to the NEP system modelling which is required to more accurately predict the impact of specific technology on system performance. The CBC model is parametrically based to allow for conducting detailed optimization studies and to provide for easy integration into an overall optimizer driver routine. The power conversion model includes the modelling of the turbines, alternators, compressors, ducting, and heat exchangers (hot-side heat exchanger and recuperator). The code predicts performance to significant detail. The system characteristics determined include estimates of mass, efficiency, and the characteristic dimensions of the major power conversion system components. These characteristics are parametrically modelled as a function of input parameters such as the aerodynamic configuration (axial or radial), turbine inlet temperature, cycle temperature ratio, power level, lifetime, materials, and redundancy.			
14. SUBJECT TERMS		15. NUMBER OF PAGES	
Nuclear electric propulsion; Closed Brayton cycle; Parametric computer modelling		160	
		16. PRICE CODE	
		A08	
17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	